Video compression algorithms for HEVC and beyond

by

André Seixas Dias

A thesis submitted to the University of London for the degree of
Doctor of Philosophy

Department of Electronic Engineering
Queen Mary, University of London
United Kingdom

July 2018

Statement of Originality

I, André Seixas Dias, confirm that the research included within this thesis is my own work or that where it has been carried out in collaboration with, or supported by others, that this is duly acknowledged below and my contribution indicated. Previously published material is also acknowledged below.

I attest that I have exercised reasonable care to ensure that the work is original, and does not to the best of my knowledge break any UK law, infringe any third partys copyright or other Intellectual Property Right, or contain any confidential material. I accept that the College has the right to use plagiarism detection software to check the electronic version of the thesis. I confirm that this thesis has not been previously submitted for the award of a degree by this or any other university. The copyright of this thesis rests with the author and no quotation from it or information derived from it may be published without the prior written consent of the author.

André Seixas Dias

24/07/2018

Details of collaboration and publications:

All papers published while working on this thesis are listed at the end of the thesis. Any publication produced in collaboration with others is clearly mentioned.

To my mother

Abstract

Due to the increasing number of new services and devices that allow the creation, distribution and consumption of video content, the amount of video information being transmitted all over the world is constantly growing. Video compression technology is essential to cope with the ever increasing volume of digital video data being distributed in today's networks, as more efficient video compression techniques allow support for higher volumes of video data under the same memory/bandwidth constraints. This is especially relevant with the introduction of new and more immersive video formats associated with significantly higher amounts of data. In this thesis, novel techniques for improving the efficiency of current and future video coding technologies are investigated. Several aspects that influence the way conventional video coding methods work are considered. In particular, the properties and limitations of the Human Visual System are exploited to tune the performance of video encoders towards better subjective quality. Additionally, it is shown how the visibility of specific types of visual artefacts can be prevented during the video encoding process, in order to avoid subjective quality degradations in the compressed content. Techniques for higher video compression efficiency are also explored, targeting to improve the compression capabilities of state-of-the-art video coding standards. Finally, the application of video coding technologies to practical use-cases is considered. Accurate estimation models are devised to control the encoding time and bit rate associated with compressed video signals, in order to meet specific encoding time and transmission time restrictions.

Acknowledgments

I would like to thank my supervisor, Prof. Ebroul Izquierdo, for giving me the opportunity to pursue a PhD in the Multimedia and Vision group and for his valuable advice. I would also like to express my deepest gratitude to my co-supervisor, Dr. Marta Mrak, for giving me the opportunity to be part of her team and for her constant guidance, support and commitment to make all this research work possible.

I am also deeply grateful to Dr. Saverio Blasi for his precious advice, support and availability to work closely with me on most of the topics addressed in this thesis. I would also like to thank Dr. Matteo Naccari for his valuable technical guidance and companionship throughout this journey and Dr. Shenglan Huang for her precious help. A big thank you also to the rest of the video compression team of BBC R&D for the friendly and positive working environment they provide.

Finally, I would like to dedicate a very special word of gratitude to my family and to all my friends that supported me during this period.

Thanks to all the above mentioned and I truly hope this work is up to your expectations.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ADZA	Adaptive Dead Zone Adjustment
AI	All Intra
AMVP	Advanced Motion Vector Prediction
ANSI	American National Standards Institute
AMP	Asymmetric Motion Partition
AVC	Advanced Video Coding
BD	Bjøntegaard Delta
CABAC	Context-based Adaptive Binary Arithmetic Coding
CAVLC	Context Adaptive Variable Length Coding
CB	Coding Block
CIP	Combined Intra Prediction
CTB	Coding Tree Block
CTU	Coding Tree Unit
CU	Coding Unit
CSF	Contrast Sensitivity Function
DPB	Decoded Picture Buffer
DCT	Discrete Co-sine Transform
DST	Discrete Sine Transform
GOP	Group Of Pictures
HD	High Definition

HDR	High Dynamic Range
HEVC	High Efficiency Video Coding
HM	HEVC test Model
HVS	Human Visual System
IEC	International Electrotechnical Commission
IBP	Inside-Block Prediction
ISO	International Organization for Standardization
ITU-R	International Telecommunication Union - Radiocommunication sector
ITU-T	International Telecommunication Union - Telecommunication sector
JCT-VC	Joint Collaborative Team on Video Coding
JM	Joint Model
JND	Just Noticeable Distortion
JVT	Joint Video Team
LD	Low Delay
MAD	Mean Absolute Difference
MOS	Mean Opinion Score
MPEG	Moving Picture Experts Group
MPI	Multi-Parameter Intra
MSE	Mean Squared Error
MV	Motion Vector
OBP	Outside-Block Prediction
PB	Prediction Block
PVP	Pixel Variation Preservation
PSNR	Peak Signal-to-Noise Ratio
PU	Prediction Unit
QP	Quantisation Parameter
QPR	Quantisation Parameter Reduction

QPMC	Quantisation Parameter reduction with Modified Costs
RA	Random Access
RAP	Random Access Point
RD	Rate-Distortion
RDO	Rate-Distortion Optimisation
RDOQ	Rate-Distortion Optimised Quantisation
RQT	Residual Quad-Tree
SAD	Sum of Absolute Differences
SAO	Sample Adaptive Offset
SOP	structure Of Pictures
SSD	Sum of Squared Differences
SSIM	Structural Similarity Index Metric
SEI	Supplemental Enhancement Information
TB	Transform Block
TU	Transform Unit
UHD	Ultra High Definition
VCEG	Video Coding Experts Group
VQM	Video Quality Metric
WPP	Wavefront Parallel Processing

Chapter 1

Introduction

The production, distribution and consumption of multimedia information keeps growing at an extraordinary pace. More and more services provide a huge variety of video content to users, who can access it almost everywhere through various types of electronic devices. Moreover, with the constant advances in consumer electronics technology, the number of new electronic devices capable of capturing, editing, storing and sharing video content all over the world is constantly increasing. This has a huge impact on the growing volume of video data conveyed in today's communication networks, which are also continuously expanding in capacity and coverage, reaching more and more people around the globe.

In addition to the higher volume of video content made available, the increasing number of new and more immersive video formats brings new challenges for storing and transmitting video. The continuous efforts to improve the user's experience through Ultra High Definition (UHD) video with higher temporal and spatial resolutions, higher bit depths or even the growing popularity of 360 degree video content are just a few examples of new features that generate a significant increase in the amount of data that needs to be handled by video services. Video compression efficiency is therefore critical in the successful deployment of new generation video formats and applications.

1.1 Problem statement and objectives of the research work

Efficient video compression technology plays a key role in the distribution of video information in countless types of video services. Typical video coding technologies achieve higher video compression ratios at the cost of higher degradations in the compressed video signal. The coding efficiency of such video compression schemes is thus determined both by the output video quality expected after compression/decompression and by the bit rate used in the encoded representation of the video signal. More efficient video coding solutions provide higher video quality for the same bandwidth/storage requirements or, alternatively, higher video quality using lower storage/bandwidth resources.

The state-of-the-art High Efficiency Video Coding (HEVC) standard [1], also known as ITU-T recommendation H.265, developed by the Joint Collaborative Team on Video Coding (JCT-VC), is able to achieve a remarkable video compression performance with respect to its predecessor, H.264/Advanced Video Coding (AVC) [2]. However, the optimisation of HEVC encoders, or even more advanced video coding tools that go beyond the capabilities of this standard, are essential to better accommodate the needs of more demanding video formats and applications.

In this context, the main objective of the work presented in this thesis is to study, design and assess new video compression tools based on the state-of-the-art HEVC standard. Altogether, these tools aim to improve the compression capabilities of future video coding solutions based on HEVC, increasing their suitability to fulfil the needs of modern video coding applications.

1.2 Thesis contributions

The work presented in this thesis as a whole addresses the need for higher compression efficiency to accommodate the needs of services relying on the transmission of video information. All techniques proposed in this work aim to provide higher efficiency to the

way video data is compressed, targeting the overall optimisation of the state-of-the-art HEVC standard from different perspectives. This is accomplished either by perceptually optimising HEVC video encoders, improving the capabilities of the standard or even enabling its seamless usage in a wider variety of practical applications where encoding time constraints need to be taken into account.

The initial part of this work comprises the development of tools that exploit the properties and limitations of the Human Visual System (HVS) with the ultimate goal of tuning the performance of HEVC encoders towards better subjective quality, rather than optimising purely according to the mathematical differences between the original and reconstructed frames after compression. Since the main objective of video communication systems is to present perceptually satisfying video information to the final user, it makes sense to optimise the compression efficiency of video compression techniques according to the perceptual properties of the HVS. In this context, a novel perceptual-based video coding technique is proposed, where a low complexity Just Noticeable Distortion (JND) model is used to drive the decisions made during the encoding process.

Another topic investigated in this thesis, that relies on understanding the HVS to improve the performance of video compression techniques, is the prevention of contouring artefatcs in compressed video. This type of artefatcs can significantly degrade the perceptual quality of the decoded video sequences, even when this degradation is not reflected in conventional objective quality metrics. Two techniques to prevent contouring artefacts in compressed videos are proposed, aiming to modify the encoding process in order to provide higher perceptual quality of the compressed content, in a fully HEVC compliant way. These techniques are proposed and evaluated in the context of HEVC video encoding, in particular to improve the output perceptual quality of compressed UHD content, which is expected to be adopted in an increasing number of video services in a near future.

In addition to perceptually-oriented optimisations, the work presented in this thesis also comprises the development of new coding tools that go beyond the capabilities of the

HEVC standard, notably through improvements to the Intra prediction process. While the previously mentioned perceptual coding tools only target the optimisation of HEVC encoders, the study on new Intra prediction tools attempts to improve the compression capabilities of the standard and therefore involves changes at the decoder side as well. This means that the bit streams produced in the context of this work are not HEVC compliant. Two techniques are proposed targeting to improve the compression capabilities of HEVC, in particular its Intra coding performance. The first technique is based on using artificial spatial patterns to improve Intra predictions while the second one is based on the so-called combined Intra prediction technique, which uses information from within a block being encoded to improve the accuracy of Intra predictions. The focus of this part of the work was on the Intra prediction process as this is a critical component of the video coding process that, compared to other essential parts of HEVC, shows more room for improvement and possibly more room to accommodate extra complexity.

Finally, the last part of this work also comprises the application of HEVC in practical video compression applications, where encoding time plays an important role. This part of the study focuses on understanding and predicting the variations in encoding time that a practical HEVC encoder may experience and how these variations can be controlled in practical scenarios. This is particularly relevant to use-cases where video content needs to be encoded and uploaded to a remote destination within a pre defined amount of time. In this context, a system that jointly controls the time spent in the encoding process and the bit rate of the generated compressed bit stream is proposed.

Each of the four aforementioned areas of research work that target the overall HEVC optimisation from different perspectives are described in four different chapters in this thesis, as detailed in the next section.

1.3 Thesis structure

The remainder of this thesis is organised as follows. **Chapter 2** provides the general background of all the conducted research work reported in this thesis. This includes the introduction of basic image and video coding fundamentals, a general description of the typical hybrid block-based video coding model and a detailed description of the state-of-the-art HEVC video coding standard, with emphasis on the most relevant tools for this thesis.

Chapter 3 presents a novel perceptual-based video coding technique, fully compliant with the HEVC standard, where a low complexity Just Noticeable Distortion (JND) model is used to drive the decisions made at the encoder. This technique exploits the characteristics and limitations of the HVS to influence the operations during the encoding process.

Chapter 4 proposes two techniques to reduce the appearance of contouring artefacts in compressed video that significantly contribute to the degradation of the visual quality of the decoded video sequences. The proposed techniques aim to provide higher perceptual quality of the compressed content by modifying the encoding process in a fully HEVC compliant way.

Chapter 5 is focused on improving the compression capabilities of HEVC, in particular its Intra coding performance. A technique based on using artificial spatial patterns to generate improved Intra predictions and a technique based on the so-called combined Intra prediction concept are proposed in this context. Both techniques require modifications both in the encoding and decoding processes and therefore are not compliant with the HEVC standard.

Chapter 6 is focused on the usage of HEVC in practical use-cases. A system is proposed to jointly control both the encoding time of a video encoder and the uploading time of the generated bit stream. The system relies on accurate encoding time and bit

rate estimation techniques in order to meet overall processing time requirements.

Chapter 7 presents some general conclusions and observations about this work, along with possible ideas for improving and expanding the proposed contributions.

Chapter 2

Background

This chapter describes the most relevant concepts behind typical video compression systems. First, some basic image and video coding fundamentals are introduced, followed by a general description of the long-established hybrid block-based video coding approach, adopted by most modern video coding standards. Finally, a more detailed description of the state-of-the-art HEVC video coding standard is given, with emphasis on the tools that are most relevant to the remainder of this thesis.

2.1 Image and video coding fundamentals

This section provides a brief description of some basic image and video coding fundamentals essential to the concepts addressed in the remaining chapters.

2.1.1 Digital image and video representation

A digital video sequence is composed of a series of still images. Each image consists of a set of samples which are displayed in two dimensions according to the spatial resolution of the sequence. For monochrome images, also commonly referred to as grayscale or

black-and-white images, a picture is represented by a single array of samples. Each sample represents the intensity value (i.e. the gray level) of the picture at the respective location. Each of these intensity values is mapped to a digital representation using a fixed number of bits per sample, referred to as the bit depth of the image. The bit depth defines the range allowed to represent intensity values. As an example, images with a bit depth of 8 bits map intensity values to a number ranging from 0 to 255, while images represented with a bit depth of 10 bits allow sample values to range from 0 to 1023.

Differently from monochrome images, the representation of colour typically requires three colour components. The impression of colour in the Human Visual System (HVS) is created due to the stimulation of photoreceptor cells by visible light refracted through the human eye. There are two major types of photoreceptor cells: cones and rods. Rods operate at low illumination levels and do not contribute for chromatic vision. Cones operate at higher light levels and are responsible for the detection of colour [3].

There are three different types of cones, characterised by their maximum sensitivity to three different wavelengths of visible light: L-cones (large wavelengths, red), M-cones (medium wavelengths, green) and S-cones (short wavelengths, blue). The perception of colour is achieved by a mixture of stimuli of these photoreceptors. Taking this into account, the representation of colour images in colour displays usually relies on the RGB color model and is typically achieved by combining three different light sources (red, green and blue) for each image element. Therefore, colour images are represented with three arrays of samples, each of them containing the intensity values of a given colour component [4].

The RGB color model is widely used both for acquiring and for displaying colour images. However, for intermediate stages such as coding and transmission/storage, colour difference signals obtained from the RGB values are preferred. These difference-based representations decouple luminance, representing the gray level intensity of the image, from chrominance, which specifies the colour information [5]. The YUV format, widely used in analog video, consists of a *luma* channel (Y) and two *chroma* channels (U and V),

obtained directly from the RGB signals after applying a given opto-electronic transfer function [6][7]. For High Dynamic Range (HDR) video, different transfer characteristics need to be considered [8].

Because the U and V signals are represented as difference signals with respect to Y, a lower bandwidth is required for their transmission. Moreover, since the HVS is more sensitive to luminance than to chrominance variations, possible transmission errors are perceptually masked by this colour representation than in plain RGB. In the case of analog television, this colour representation also allowed backward compatibility between black-and-white and colour television when the latter was introduced.

In the case of digital image and video coding, YCbCr is the preferred colour representation. In this colour representation, Y also corresponds to the luma component while Cb and Cr denote the blue-difference and red-difference chroma components. All these components are obtained from the previously mentioned YUV components after applying appropriate scaling and offset operations to place the signals into a digital form [6][7]. Overall, the YCbCr colour representation can be obtained by a mathematical coordinate transformation from the corresponding plain RGB colour representation, making it trivial to switch between the two and consequently allowing using different representations for different purposes. Since the main focus of this thesis is on video coding-related techniques, YCbCr will be the colour representation assumed for image and video signals throughout the rest of this document. An example of the RGB and YCbCr colour representations is shown in Figure 2.1.

As previously mentioned, the HVS is more sensitive to brightness variations than to colour. It is therefore possible to take advantage of this property to reduce the amount of information needed to represent an image or video by reducing the spatial resolution associated with chroma components. This is denoted as chroma sub-sampling and the notation used for three of the most common formats adopted for this purpose is the following:

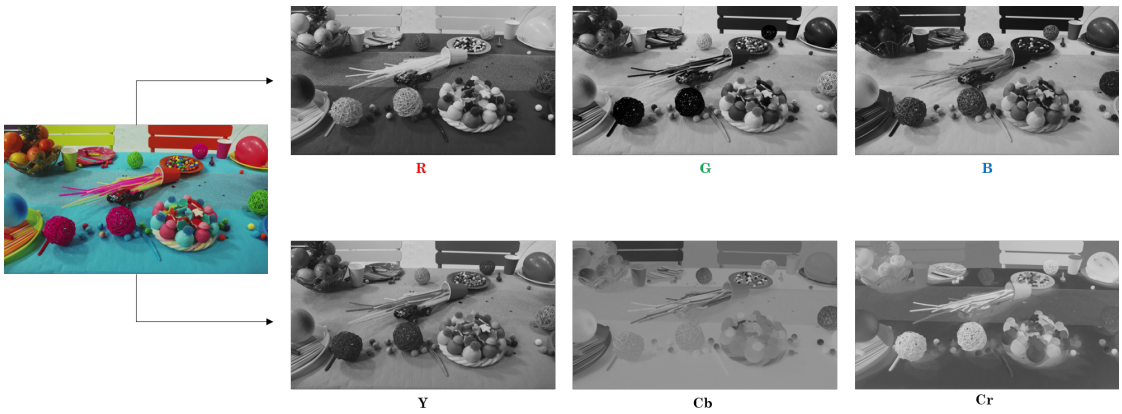


Figure 2.1: RGB and YCbCr colour representations.

- **4:4:4** - No chroma sub-sampling is applied, meaning that both luma and chroma components have the same spatial resolution.
- **4:2:2** - Both chroma components are sub-sampled to half the size of the luma resolution horizontally. Vertically, the resolution of the chroma components is the same as luma.
- **4:2:0** - Both chroma components are sub-sampled to half the size of the luma resolution, both horizontally and vertically.

Figure 2.2 shows a graphical representation of the described colour sub-sampling formats. The 4:2:0 scheme is the preferred format for distribution of video and therefore it is the most commonly used in video coding applications targeting distribution. The 4:2:2 format is preferred for production and contribution scenarios [9].

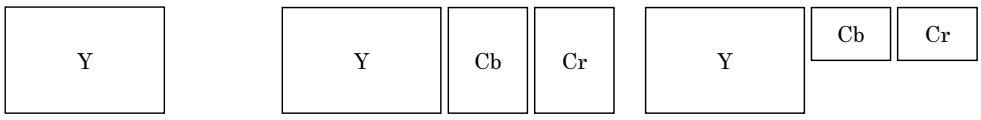


Figure 2.2: Most common chroma sub-sampling formats. a) 4:0:0; b) 4:2:2; c) 4:2:0.

Considering the temporal aspect of video, the pictures composing a video sequence,

also referred to as frames, are sequentially displayed at a given frame rate. The frame rate is usually measured in Hertz (Hz), with 1 Hz = 1 s^{-1} or, equivalently, 1 frame per second (fps). In order to create the sensation of continuous motion for human viewers, the frame rate at which pictures are displayed needs to be, in general, higher than around 24 fps [9]. However, for content with a high amount of motion and camera panning, such as sports, capturing the content at higher frame rates provides, in general, a better representation of the scene. The High Definition (HD) television format, for example, supports frame rates of up to 60 fps while Ultra High Definition (UHD) television standards support up to 120 fps. For some types of content and higher spatial resolutions, it is likely that frame rates even higher than these may contribute to an overall improved quality of experience [10] [11].

Along with the temporal resolution, the spatial resolution of a video sequence also plays an important role in the perceived quality of video signals. In modern digital television broadcasting, for example, the HD television format [6] uses a spatial resolution of 1920×1080 pixels, while in the UHD format [7], spatial resolutions of 3840×2160 or 7680×4320 pixels are supported. These spatial resolutions are commonly referred to as 4K UHD and 8K UHD, respectively. The higher the temporal and spatial resolutions associated with video signals, the higher the resources needed in terms of transmission bandwidth/storage to support the distribution of video. For this reason, video compression technology plays an important role in the deployment of such high resolution video formats.

2.1.2 Video coding systems

The typical workflow of a video encoding-decoding system is depicted in Figure 2.3. Typically, the captured video content is fed to a video encoder to produce the encoded bit stream, which is a more compact representation of the video information and therefore more adequate for transmission. The bit stream is then transmitted to its destination through a transmission channel, which can be, for example, a transmission network, a

radio link, distribution through storage devices, and so on. At the receiver end, the compressed bit stream is fed to the video decoder, where the video content is reconstructed and displayed to the final user.

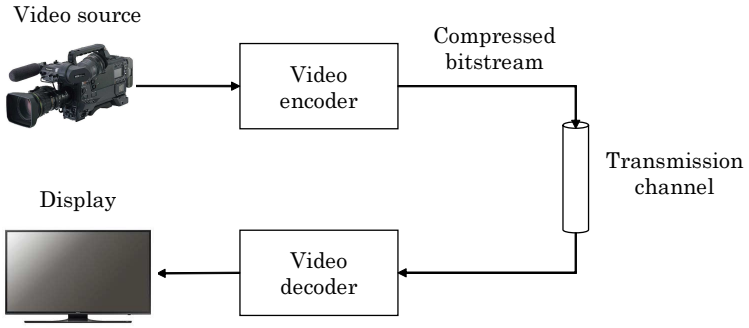


Figure 2.3: Typical video encoding and decoding workflow.

For the workflow in Figure 2.3 to work, video encoders need to produce a bit stream that decoders can understand and decode to generate the corresponding video output to display. In order to avoid the coexistence of several different compression schemes each requiring its own decoding process, many efforts are allocated to the specification of video coding standards. A video coding standard typically consists of a precise specification of the syntax and semantics of a bit stream, along with the precise steps required to perform the decoding process and produce a reconstructed video sequence. This means that standards define the tools that can be used to compress a video signal, as only these tools are guaranteed to be correctly interpreted by standard-compliant decoders.

Video coding standards are essential to guarantee the interoperability between a wide variety of devices developed by different manufacturers for similar purposes. It is important to emphasise that in video coding standards, only the decoding process is standardised, as standardising the encoding process is not required for interoperability. In fact, each encoder manufacturer has the freedom to independently use the tools provided by the standard to achieve its target coding performance, as long as a compliant bit stream is produced by a given encoder.

The video encoding process exploits the temporal, spatial and statistical redundancy

present in the video content to reduce the information needed to represent the video signal. Additionally, in most practical video coding applications, irrelevance also needs to be exploited to achieve the desired compression ratios. By doing so, some visual information in the original video content is lost after the encoding process, as the original video content cannot be fully reconstructed at the decoder side. This draws an important distinction between two possible types of video compression: lossless and lossy. Lossless compression occurs when only redundancy is exploited and therefore there is a perfect match between the original and decoded video sequences. Conversely, lossless compression also exploits irrelevance, making the encoding process irreversible.

Lossless compression is used in applications such as medical imaging, where visual information losses are not acceptable. On the other hand, even the highest compression ratios achieved with lossless compression techniques are not sufficient for applications like television broadcasting, web streaming, social media and so on. In these cases, lossy compression is essential to cope with the storage and bandwidth limitations imposed by the infrastructure used to distribute the content.

2.1.3 Performance measurements

As mentioned in the previous subsection, when lossy compression is applied, the reconstructed signal output by the decoder is different from the original signal. Such difference is usually quantified by measuring the objective distortion of the reconstructed pictures with respect to the original ones. This objective distortion if often used as an indication of the quality of the reconstructed pictures. Typical compression algorithms achieve higher compression ratios at the cost of higher distortions. The coding efficiency of a lossy video compression scheme is thus determined both by the output video quality expected at the receiver and by the bit rate used to convey the encoded bit stream.

2.1.3.1 Distortion metrics

The purpose of a distortion metric is to provide a numerical score to evaluate the quality of the reconstructed signal with respect to the original one. The most commonly used distortion metrics in image processing applications and, in particular, in video coding technologies, are summarised in the following. These metrics are computed assuming an original signal $x(i,j)$ and a reconstructed signal $x_{rec}(i,j)$ where $i = 0, ..., N-1$ and $j = 0, ..., M-1$ represent the vertical and horizontal indices of each sample, respectively, in a given $N \times M$ image area (e.g. luma samples in a given block of the image or in the whole frame).

- **Sum of Absolute Differences (SAD)** - The SAD can be considered the simplest similarity metric used to quantify distortions in an image. The SAD is given by

$$SAD = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} |x(i,j) - x_{rec}(i,j)| \,. \tag{2.1}$$

  The difference signal can be seen as noise which was added to the original signal. The SAD corresponds to the L1 norm in a vector space.

- **Sum of Squared Differences (SSD)** - The SSD, also referred to as Sum of Squared Error (SSE), is a distortion metric widely used in signal processing applications, given by

$$SSD = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} [x(i,j) - x_{rec}(i,j)]^2 \,. \tag{2.2}$$

  The SSD corresponds to the L2 norm of the difference signal, which represents the energy of the noise added to the original signal.

- **Mean Squared Error (MSE)** - Following the definition of the SSD, the MSE represents the average energy of the difference signal, given by

$$MSE = \frac{SSD}{N \cdot M}. \tag{2.3}$$

- **Peak Signal to Noise Ratio (PSNR)** - Finally, the PSNR is the most commonly used objective distortion metric for video quality assessment in the context of video coding schemes. It expresses the ratio in dB between the peak value of the signal and the MSE. The PSNR is formally given by

$$PSNR = 10 \cdot \log_{10} \left(\frac{A_{max}^2}{MSE} \right), \tag{2.4}$$

  where A_{max} represents the highest possible intensity value of a sample, i.e. $A_{max} = 255 = 2^8 - 1$ for 8-bit video and $A_{max} = 1023 = 2^{10} - 1$ for 10-bit video. Higher PSNR values correspond to lower MSE and are therefore associated with better quality of the distorted signal with respect to the original one.

Objective quality metrics are very useful for the development of video compression tools. PSNR, in particular, is widely used both to drive decisions during the encoding process and to evaluate the output video quality of the reconstructed images. However, PSNR and other similar per-pixel distortion metrics do not necessarily correlate well with the way humans perceive visual information (see Figure 2.4). Since they are exclusively based on the mathematical differences between original and reconstructed frames, these metrics do not account for the complex mechanisms inherent to the HVS that define the perception of visual quality [12]. In fact, these mechanisms are so complex that a fully reliable visual quality evaluation metric or scheme is yet to be defined [13].

Subjective video quality evaluation is therefore the most reliable way to assess video quality. However, conducting formal subjective evaluations is often an expensive and time-consuming task, which is why objective metrics are preferred in most cases for the development of video coding algorithms.

Several objective video quality metrics were proposed in the literature targeting to provide higher correlation with human visual perception than PSNR. The Structural Similarity Index Metric (SSIM) [15], for example, takes into account the fact that the so-called structural distortions, such as additive noise, blur, blocking artefacts and so on,

(a) (b)

Figure 2.4: Visual comparison between two images with the same PSNR value (same amount of noise introduced). a) Noise randomly introduced; b) Noise introduced in specific parts of the image [14].

can be expected to have a stronger impact on the subjective quality of the signal. On the other hand, spatial shifts, contrast variations and similar distortions are expected to have a lower impact. The SSIM metric has gradually become more popular as an additional metric used for still image quality assessment.

Another example is the Video Quality Metric (VQM) [16], standardised as a recommended video quality metric by standardisation bodies such as the International Telecommunications Union (ITU) [17][18] and the American National Standards Institute (ANSI) [19]. VQM takes into account different features of a video sequence that have an impact on the perception of quality, such as the amount of motion, spatial gradients and contrast information.

It is worth noting that PSNR and MSE are still the most used distortion metrics in video coding research. This is mostly due to the fact that despite their non-optimal characteristics, these metrics are particularly well suited to measure video quality in most of the scenarios of interest for video coding research. The study in [20] highlights the fact that MSE performs well when used to compare the same content encoded with the same codec, in order to validate the performance when different tools are enabled or

disabled.

2.1.3.2 Rate-distortion theory

Modern video coding schemes operate by selecting the best set of coding options in order to efficiently compress the source signal into a bit stream. The size of a bit stream is usually measured in terms of the average number of bits necessary to encode a second of video (in bits per second), and is usually referred to as the bit rate. As mentioned before, most schemes achieve higher compression ratios (i.e. lower bit rates) at the cost of larger distortions. An efficient encoder should be capable of selecting and tuning its tools based on a trade-off between bit rate and distortion. This process is commonly referred to as Rate-Distortion Optimisation (RDO) [21].

Most of the techniques used for RDO are based on Lagrangian optimisation [21]. In particular, assume that the encoder is currently selecting which tool, out of a set of possible tools T_k, $k = 0, ..., N$, should be used to encode the currently considered part of the input signal. Each tool takes the input signal $x(i)$ and produces the corresponding compressed bit stream with bit rate b_k which, once decoded, results in a reconstructed signal $x_{rec,k}(i)$. Define as $D\left(x(i), x_{rec,k}(i)\right)$ a certain distortion metric, such as those described in the previous subsection. The Lagrangian cost associated with tool T_k is defined as:

$$J\left(T_k\right) = D\left(x(i), x_{rec,k}(i)\right) + \lambda \cdot b_k, \tag{2.5}$$

where λ is the Lagrangian multiplier. The optimal tool T^o can be selected by minimising the aforementioned cost, according to:

$$T^o : \min_{T_0,...,T_N} \{J\left(T_k\right)\}. \tag{2.6}$$

Most modern video encoders use this type of RDO to achieve the desired compression performance. It is important to note that the best trade-off given by the RDO process of a video encoder depends on the target quality or bit rate. The same video compression scheme may target very low distortion, usually producing a high bit rate as a consequence, or a very low bit rate, in which case higher distortions are likely to occur.

In order to evaluate the performance of a video compression scheme for different purposes, a range of operation points is considered. The distortion of the reconstructed signal and the bit rate of the compressed bit stream are measured for each of the selected operation points. The results of such tests are usually visualised in the form of the so-called Rate-Distortion (RD) curves, consisting of plots in which distortion values are plotted against the corresponding bit rates for each operation point. RD curves can be used to compare the efficiency of different coding schemes, in particular to compare the performance of new video compression tools against a given benchmark, often denoted as anchor. An example of such a comparison is shown in Figure 2.5 using PSNR as distortion metric and 4 different quality/bit rate operation points.

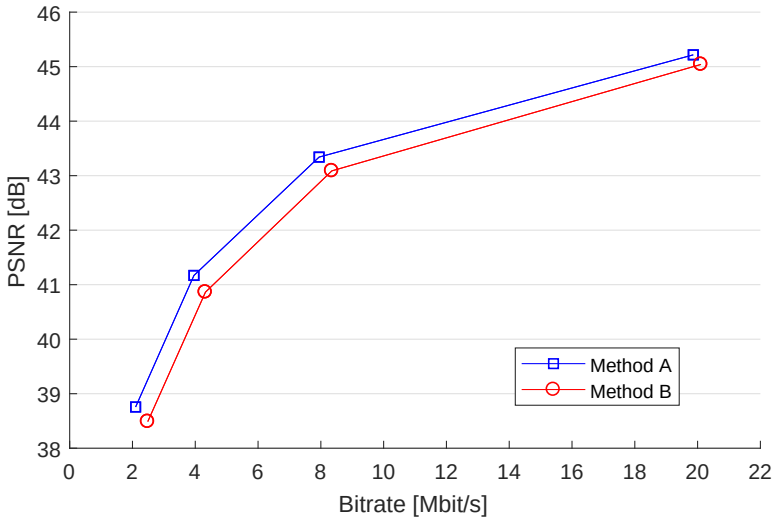


Figure 2.5: Example of the RD curves of two different video compression schemes.

For the case in Figure 2.5, it is clear that A shows superior RD performance than

B. This is because the blue curve is above the red curve, meaning that for the same bit rate, A is able to provide higher decoded video quality (less distortion). From a different perspective, it is also possible to consider that A is able to achieve the same decoded video quality as B using lower bit rates for the whole range of points tested.

2.1.3.3 The Bjøntegaard delta metrics

Contrarily to the comparison of coding technologies represented in Figure 2.5, the RD curves associated with two different video compression techniques are not always clearly separated from each other. In these cases, it is difficult to determine which coding scheme performs better and to quantify the associated performance difference. For this reason, the Bjøntegaard model [22] has become a popular tool for evaluating the coding efficiency of a given video codec in comparison with a reference codec over a range of quality points or bit rates.

Bjøntegaard Delta (BD) metrics are typically computed as a difference in bit rate or difference in quality based on interpolating curves from the tested data points. The difference in bit rate, denoted as BD-rate, is the most used in the literature and it is expressed as a percentage of a reference bit rate. This percentage represents the average bit rate savings for the same video quality (e.g. measured with PSNR) and is calculated between two rate-distortion curves, such as the ones in Figure 2.5.

Since the BD-rate is represented as a percentage of a reference bit rate, negative BD-rate values represent compression gains, while positive values represent compression losses. In short, the BD-rate is obtained by computing the difference between the area below the inverse of the RD curves obtained by the two schemes being compared (i.e. the anchor and the test). The larger this difference is, the better the method with respect to the benchmark in a rate-distortion sense. This area is computed for a given quality overlapping interval, highlighted in Figure 2.6.

The Bjøntegaard model uses a logarithmic scale for the domain of the bit rate inter-

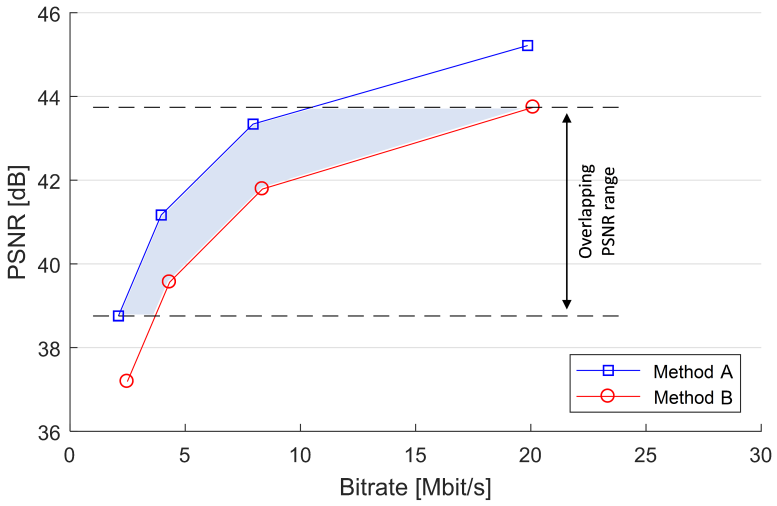


Figure 2.6: Example of quality interval used in BD-rate computation.

polation and the curve fitting is often obtained using a cubic spline interpolation. The detailed description of the BD-rate computation can be consulted in [22].

2.2 The hybrid block-based video coding model

During the past three decades, the most well-known and worldwide used video coding standards have been developed based on a hybrid block-based video coding model. This type of coding approach has been adopted by both the ITU-T H.26x and the ISO/IEC MPEG-x families of video coding standards since the early ITU-T H.261 [23] until the most recent (joint ITU-T and ISO/IEC MPEG) HEVC standard. Although there is a large difference between all these standards regarding the coding efficiency and also associated complexity, the basic principles and tools behind them are essentially the same. This coding model is called hybrid as it combines temporal prediction between pictures of the video sequence with transform coding techniques for the prediction error [24]. Figure 2.7 shows a simplified block diagram of the functional architecture of a block-based hybrid video encoder.

Following the diagram in 2.7, a typical block-based hybrid video encoder comprises

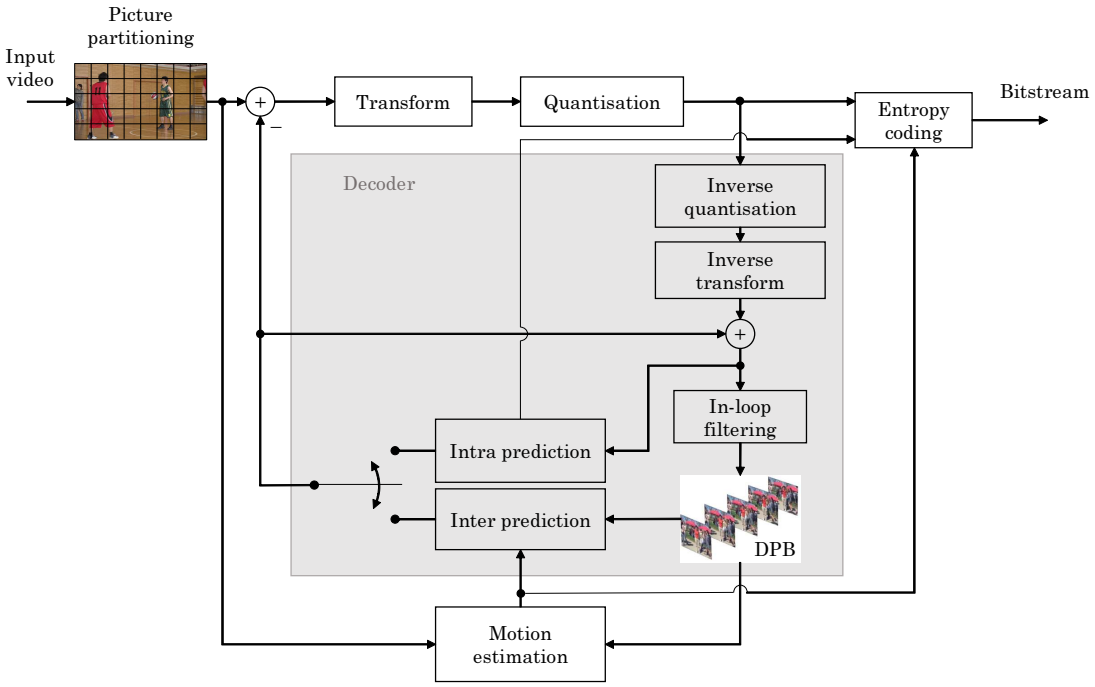


Figure 2.7: Functional architecture of a typical block-based hybrid video encoder.

the following operations:

- **Picture partitioning** - The original input video signal is partitioned into non-overlapping blocks of a certain size, which are processed separately. Each block is then encoded based on the idea that its content can be predicted from other parts of the signal, due to the presence of redundancy. The prediction signal is generated using information available both at the encoder and decoder and can be of the type Intra or Inter.

- **Intra prediction** - The Intra prediction process aims to generate a prediction of the current block being encoded using information from previously encoded parts of the same frame. The basic assumption is that neighbouring samples in a frame are typically correlated and therefore the content of a given block can be predicted using information from the already encoded samples in its spatial neighbourhood, exploiting the so-called spatial redundancy. Predictions are typically obtained by displacing already encoded neighbouring samples (denoted as reference samples)

into the current block being predicted, according to a given direction. This direction, along with any other information used to build the prediction, needs to be conveyed in the bit stream, so that the decoder can generate the same Intra prediction signal.

- **Inter prediction** - The Inter prediction process, on the other hand, seeks to exploit the temporal redundancy between different frames of a video signal. The content of successive frames is typically similar and therefore only small differences between frames occur, usually due to motion. A motion compensated Inter prediction is generated using motion information extracted for this purpose.

- **Motion estimation** - Since different parts of previously encoded frames can be used for Inter prediction, the encoder needs to search for the most appropriate ones to use. This process is referred to as the motion estimation process and essentially consists of taking the current block and trying to find the best matching area in a previously encoded frame. Once this area is found, the information on how to obtain the selected prediction is used to generate the motion compensated prediction for the current block. This information includes, for example, the index of the previously encoded frame used as reference, called the reference frame index, and a Motion Vector (MV) indicating the position of the prediction block in the reference frame relative to the position of the current block. This motion information also needs to be conveyed in the bit stream, so that the same Inter prediction block can be generated at the decoder side. It is important to note that the motion estimation process is typically only performed at the encoder side, as only the encoder has access to the original video signal. The decoder only extracts the motion information from the bit stream and consequently performs motion compensation to compute the prediction for a given block.

- **Transform** - Using the generated Intra or Inter prediction signal, the difference between the original input block and the prediction signal is computed to create the residual signal. The residual signal contains the part of the original signal

that could not be predicted by the selected prediction method. This signal is then transformed to a different representation, in order to achieve further decorrelation. The transform process produces a set of coefficients able to describe a given residual block in the frequency domain, typically concentrating the information present in its samples into a smaller number of values.

- **Quantisation** - The coefficients produced by the transform process are then quantised according to a given quantisation step. Scalar quantisation is typically applied and each transform coefficient is independently quantised. This process maps the amplitudes of the transform coefficients into a predefined set of representative values. This is the part of the encoding process that introduces losses, as after quantisation, the original signal cannot be recovered. Finer quantisation (small quantisation step) introduces lower losses than a coarser quantisation (large quantisation step). Therefore adjusting the quantisation step controls the expected target bit rate/quality.

- **Entropy coding** - The quantised transform coefficients, also referred to as quatisation levels, along with the Intra and Inter prediction information, are then entropy encoded to generate the bit stream. The entropy encoding process is a lossless operation that further exploits the statistical redundancy in the coding elements, taking into account their probability of occurrence. At the decoder side, the inverse operation is performed to retrieve the entropy encoded information.

- **Inverse quantisation and inverse transform** - In order to synchronise its operation with the decoder, the encoder performs the inverse quantisation and inverse transform operations. This is needed because the decoder does not have access to the original signal and therefore generates its predictions based on previously encoded parts of the video sequence. This forces the encoder to replicate the decoding operation by adding the inverse transformed residual to the current prediction signal, creating the reconstructed frames that will be used as reference for subsequent predictions.

- **In-loop filtering** - After finalising the encoding of all blocks of a given frame, one or more filtering processes are applied to the recently encoded frame. These filters include, for example, a deblocking filter, used to minimise the visual effects of discontinuities across block boundaries. All in-loop filtering processes applied at the encoder and decoder are identical in order to keep the synchronism in terms of internal representations of previously encoded frames. The filtered frames are stored in the so-called Decoded Picture Buffer (DPB) to be used as reference for subsequent frames, hence the name "in-loop". The frames stored in the DPB are denoted as reconstructed frames and are exactly the same as the decoded frames output by the decoder, assuming that no transmission errors occur between encoder and decoder.

The block-based hybrid video coding model has proved to be well-suited for efficient coding and decoding of video content. Even though the diagram in Figure 2.7 intends to show the typical operation of a video encoder, the operation of the decoder can be visualised in the gray box highlighted in the same figure. This is because, as previously explained, the decoding operation is replicated at the encoder side to keep the synchronism between internal representations of reconstructed frames.

One final important characteristics of the hybrid video coding model is the complexity allocation. Both temporal and spatial redundancy exploitation are performed at the encoder. This means that the encoder needs to analyse the video content and perform the necessary tests to appropriately select the best way to code each block of the input video signal. The decoder is much simpler, as it only needs to interpret the encoded bit stream to reconstruct video frames, following the choices previously made by the encoder. This asymmetrical complexity allocation is very well-suited for applications following the so-called down-link model, such as digital television broadcast and digital video storage, where video source signals are encoded by a few encoders and decoded by millions of decoders.

2.3 The High Efficiency Video Coding standard

HEVC is the state-of-the-art video compression standard, developed jointly by the ITU-R VCEG and the ISO/IEC MPEG. The standard is published by the ITU-T as recommendation H.265 and by ISO/IEC as MPEG-H part 2. It is commonly referred to as HEVC, which is the name used throughout this thesis. The first version of the standard was approved in January 2013 and version 2 followed the year after, adding scalable, multiview and range extensions profiles [25]. Version 3 was published in 2015 adding the 3D Main profile [26], version 4 added profiles for screen content coding in 2016 [27] and finally in 2018, version 5 added support for Supplemental Enhancement Information (SEI) messages related to omnidirectional video and introduced additional profiles, such as the Main 10 Still Picture profile.

The need to create HEVC as a new standard for video compression stemmed mainly from the wide availability of high definition video content and displays, the emerging possibility of new UHD video services and the increasing deployment of high-quality video over mobile devices. The development of the new standard focused on providing substantially improved compression efficiency with respect to its predecessor, AVC, taking into account higher resolution video formats and allowing increased use of parallel architectures [28].

Similarly to its predecessors, HEVC follows the block-based hybrid model described in the previous section. Despite having a similar architecture to AVC, almost all aspects of the coding process were improved and optimised, allowing it to provide around 50% higher efficiency than AVC [29]. HEVC is also considerably more efficient than state-of-the-art still image coding standards such as JPEG2000 [30], for example, with 44% higher efficiency reported on average [31].

This section describes in detail the most relevant aspects of the HEVC standard, with focus on the key features that differentiate it from previous standards of the same family.

2.3.1 Coding structures

One of the key factors responsible for the high efficiency of HEVC is the adopted spatial partitioning structure, which provides higher flexibility to the encoding process. The base entity responsible for this partitioning is the Coding Tree Unit (CTU). A CTU is a syntax structure that identifies a square area of $N \times N$ samples in the frame and determines how this area is possibly partitioned into smaller units, called Coding Units (CUs). Each CU can be Intra or Inter-coded, as described in the following subsections.

HEVC allows a maximum $N = 64$ for CTUs. Each CTU can then be partitioned in a quad-tree fashion into smaller CUs. This is indicated by a binary split flag present at the beginning of each CTU indicating whether the original $N \times N$ unit is coded as a single CU, or if it is split into four smaller CUs, each covering an image area of $N/2 \times N/2$ samples. In the second case, each of the resulting smaller CUs starts with a binary split flag, indicating if this CU is coded as a whole or if it is further split into four $N/4 \times N/4$ blocks. This process continues until all CUs within the CTU are not partitioned any further or until the minimum CU size specified is achieved. The minimum CU size allowed in HEVC is 8×8. Figure 2.8 shows an example of the spatial partitioning of a given CTU and the corresponding coding quad-tree.

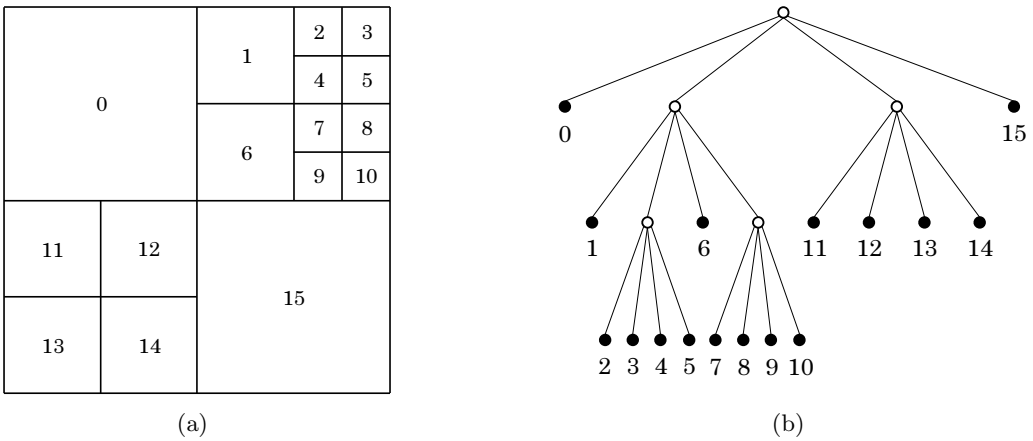


Figure 2.8: Spatial partitioning in HEVC: a) Coding Tree Unit; b) Respective coding quad-tree.

Each CU inside the CTU is assigned a depth depending on its size: depth 0 for the largest $N \times N$ CU, depth 1 for $N/2 \times N/2$ CUs and so on. As HEVC allows a minimum CU size of 8×8 luma samples, when $N = 64$ the CTU can split the original largest CU into smaller CUs up to a maximum depth of 3.

It is important to distinguish the terminology used in the HEVC specification regarding blocks and units. The term "block" refers to a specific area in a sample array (e.g. luma), whereas the term 'unit' refers to the collocated blocks of all encoded colour components (luma and chroma) and associated syntax elements, including prediction information such as motion vectors. Thus, a given CTU contains the corresponding Coding Tree Blocks (CTBs) of all encoded colour components along with the respective prediction information, forming a complete entity in the bit stream syntax. The same relationship applies to a CU and its Coding Blocks (CBs).

The fact that each frame can be partitioned in blocks of variable size possibly larger than the size of macroblocks used in AVC is crucial to the efficiency of HEVC. In fact, restricting the maximum CU size to $N = 16$ (the same size as that of the macroblocks in AVC) considerably affects the performance of HEVC. A decrease in performance of up to 30% in BD-rates was reported in some cases when imposing such a restriction during the encoding [32].

Regarding the temporal structure, in HEVC and in previous standards, frames can be encoded in a different order than the display order, depending on the desired encoding configuration. Three encoding configurations are particularly relevant, referred to as Low Delay (LD), Random Access (RA) and All Intra (AI).

In the LD configuration, frames are encoded in display order. Only frames from temporally past instants can be used as reference for inter-prediction. This configuration is relevant, for example, for real-time video conferencing applications, where encoding and decoding delays are prohibitive.

The RA configuration uses a more complex coding order where frames are encoded

in groups, commonly referred to as Intra periods, each of them starting with a Random Access Point (RAP) frame. A decoder can start decoding a bit stream from any RAP frame, which means that the frames after a RAP frame cannot depend on the content in any frame before this RAP frame. Additionally, within an Intra period, a so-called hierarchical coding structure is typically used, where smaller groups denoted as Structures of Pictures (SOP) are defined. The SOP defines specific encoding parameters, such as the encoding order of its frames (which may be different than their display order, referred to as the Picture Order Count, POC), the reference frames used for inter prediction, and so on. Due to this higher flexibility for temporal prediction, the RA configuration provides, in general, better compression efficiency than the LD configuration if the same Intra period is used.

Finally, in the AI configuration, all frames are coded without any reference to previously encoded frames, i.e. only Intra prediction is used. This configuration is relevant for situations where no Inter-frame dependencies are allowed. In this thesis, most of the proposed algorithms were developed and tested targeting a RA configuration, with the exception of the Intra coding tools described in Chapter 5, where an AI configuration is used.

Finally, the aforementioned temporal coding structures are achieved by specifying the type of slice used to encode each frame of the input video sequence. A slice is a sequence of CTUs and in HEVC, the following three slice types are supported:

- **I slices** - Only allow CTUs to be encoded using Intra prediction.
- **P slices** - Allow CTUs to use Intra and unidirectional Inter prediction.
- **B slices** - Allow CTUs to use Intra and both unidirectional and bi-directional Inter prediction.

A frame can be encoded using one or multiple slices. Slices are usually associated with the packetisation scheme used at a higher level to transmit the bit stream. Multiple

slices per frame are useful, for example, when transmitting compressed bit streams over unreliable transmission channels where packet losses may occur. For the cases addressed in this thesis, a single slice is used to encode each frame of the video. For this reason, the terms "frame" and "slice" may be used interchangeably unless otherwise specified.

2.3.2 Intra prediction

For a given CU belonging to the coding quad-tree structure described in the previous subsection, the corresponding image area of the content is predicted using one or more Prediction Units (PUs). For Intra-coded CUs, a single PU of the same size of the entire CU is generally used to perform the prediction. This PU mode is referred to as Intra-$2N \times 2N$. For the special case of the minimium possible CU size, an additional partitioning into four smaller PUs may be applied. This mode is referred to as Intra-$N \times N$. Both Intra prediction modes used in HEVC are shown in Figure 2.9. Given the minimum CU size of 8×8 allowed in HEVC, the minimum PU size that can be independently Intra-predicted is 4×4.

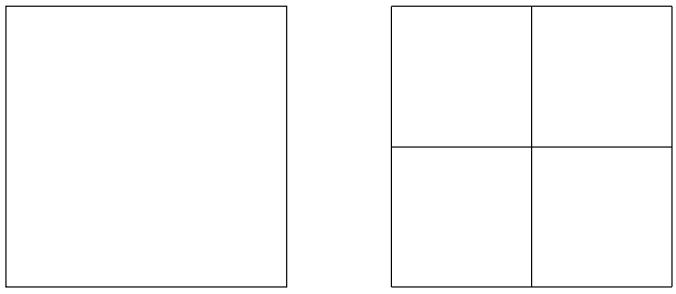


Figure 2.9: Intra PU modes available in HEVC.

At the PU level, three types of Intra prediction are available, namely DC, planar and angular prediction modes. For the latter, up to 33 angular prediction directions are allowed to predict the luma component. This is considerably more than the number of modes available in AVC, allowing higher prediction accuracy than the previous standard.

The available Intra prediction types are illustrated in Figure 2.10(a), along with the previously reconstructed samples used as reference to build the predictions in Figure 2.10(b). Dashed arrows in Figure 2.10(a) correspond to angular predictions that involve interpolation of reference samples. Conversely continuous arrows correspond to modes that involve simply copying reference samples into the predicted block, as explained further in this subsection.

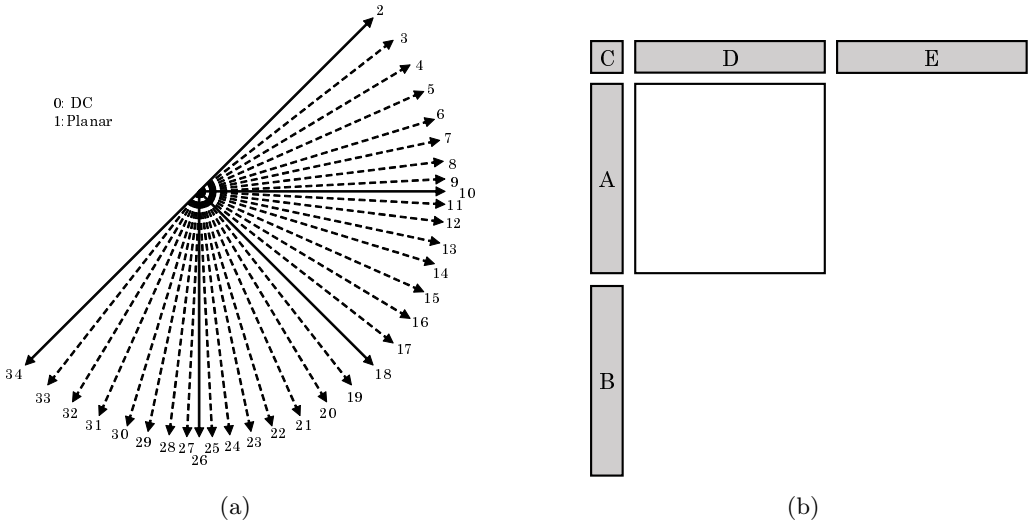


Figure 2.10: a) Intra prediction types; b) Reference samples used for Intra prediction in HEVC (A-E).

The content of a PU is predicted using one Intra prediction mode selected from the list of available modes. It is important to point out that, while the same Intra prediction mode is used to predict all samples within a PU, the process of computing the actual Intra prediction is not performed at a PU level. As explained further in this section, each PU can be further partitioned into square blocks, referred to as Transform Units (TUs) for residual coding purposes. Intra prediction is performed separately for each TU within a PU, using different reference samples. Thanks to this method and due to the particular way in which a block is partitioned in TUs, a larger number of reference samples can be used for Intra prediction. However, not all reference samples are always available for all TUs; samples that are not available are either not considered or replaced with predefined values [33].

Additionally, a simple three-tap filter is applied in HEVC to the reference samples used for Intra prediction in particular modes and TU sizes. Filtering the reference samples prior to Intra prediction has the goal of distributing more smoothly the information in such samples, consequently spreading the residual error more uniformly in the predicted block [34].

DC Intra prediction is performed by computing the average of the values of the adjacent horizontal and vertical neighbour reference samples (samples A and D in Figure 2.10(b)). This average value is then assigned to all samples of the current prediction block, as illustrated in Figure 2.11(a) for a 4×4 block, where the average value is denoted as v_{DC}. A simple filtering process is applied in the borders of the predicted block to smooth the transition between reference samples and predicted samples in these areas.

For the planar Intra prediction type, the prediction of a sample in a block is generated by a weighted average of four reference samples. An example of the reference samples used for this purpose is illustrated in Figure 2.11(b) also for a 4×4 block. The weight assign to each reference sample involved in the computation depends on the location of the sample being predicted. Closer reference samples are assigned higher weights while reference samples further away are assigned lower weights [33].

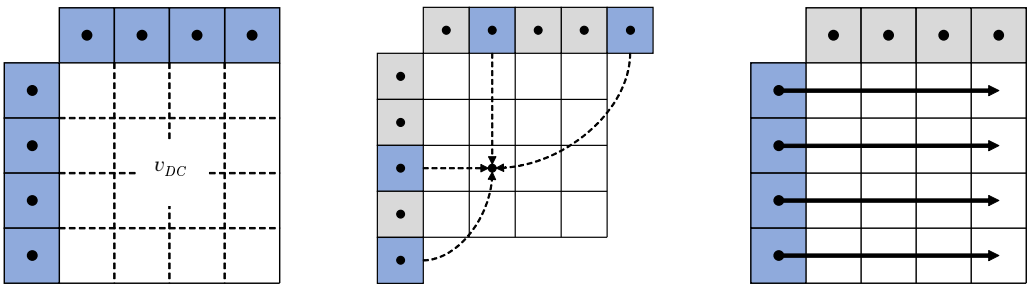


Figure 2.11: Different Intra prediction types. a) DC; b) Planar; and c) Angular (horizontal, mode 10)

As for the angular prediction modes, the Intra predicted samples are given by a

displacement of the reference samples in the direction defined by one of the 33 prediction angles illustrated in Figure 2.10. An illustration of the horizontal prediction process (mode 10) is illustrated in Figure 2.11(c). For the cases of horizontal, vertical and the two diagonal prediction angles, the values of the reference samples are directly copied into the block being predicted to form the prediction. For the remaining cases, since the prediction angles require reference samples from non-integer sample positions, a linear interpolation is performed between the two closest reference samples using 1/32 sample accuracy [33].

HEVC allows 35 possible modes for each luma PU. To limit the bits needed to signal the choice of Intra prediction mode for the current PU, a list of 3 most probable modes is considered using available information such as the Intra prediction modes found for the PUs on top and left side of the current PU (if they are available). If the Intra prediction mode chosen for the current PU is inside the list, an index is transmitted in the bit stream to signal which element in the list is used. Otherwise, the Intra prediction mode is fully signalled using a fixed codeword of 5 bits.

Chroma Intra prediction is performed after the prediction of luma. To limit complexity and also reduce the number of bits needed to signal the chroma Intra prediction mode, only up to 5 Intra prediction modes are allowed. Chroma Intra prediction is forced to use the same Intra prediction mode used for luma in case this mode is 0 (DC prediction), mode 1 (planar prediction), mode 10 (pure horizontal angular prediction) or mode 26 (pure vertical angular prediction). No additional information is transmitted in the bit stream in these cases. Otherwise, if luma samples are predicted with a mode different than 0, 1, 10 or 26, these modes are all considered for chroma prediction along with a fifth mode (referred to as derived mode) that is set equal to the one used for the luma component.

In Chapter 5, some additional modifications to the HEVC Intra prediction process described here are proposed. These modifications are still based on a similar workflow, using additional elements to build more accurate Intra predictions, as further explained.

2.3.3 Inter prediction

A CU can be partitioned in PUs for Inter prediction in 8 different ways. In mode $2N \times 2N$, the full CU is predicted as a single PU of the same size. In modes $2N \times N$ and $N \times 2N$, two PUs are created by splitting the CU in two identical parts of half the height or half the width of the CU, respectively. Mode $N \times N$ corresponds to partitioning the CU into four PUs of identical size. Finally, HEVC supports 4 additional Inter prediction modes, referred to as Asymmetric Motion Partition (AMP) modes, where the CU is split asymmetrically in two PUs. Figure 2.12 illustrates all these possible partitions of a CU for Inter prediction.

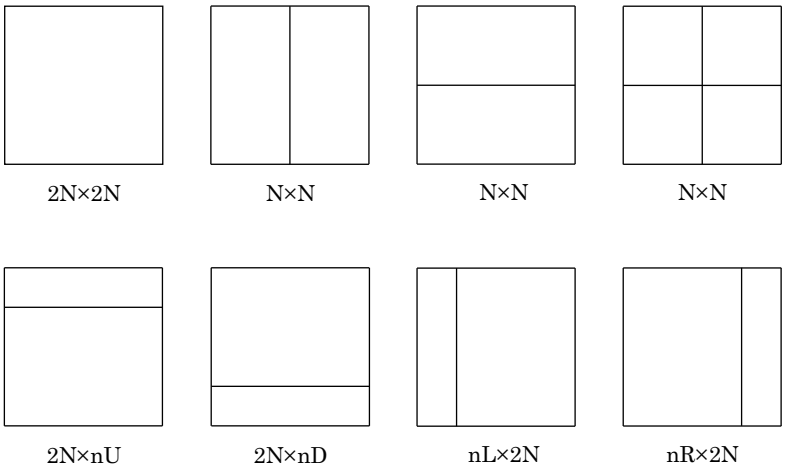


Figure 2.12: Inter-prediction modes in HEVC.

Each PU is independently predicted using one or more previously encoded reference frames. Up to two lists of reference frames (denoted as List0 and List1) may be considered depending on the slice type. For PUs in P-slices, only reference frames included in List0 can be used for prediction. Only uni-prediction is available in this case, meaning that only a single prediction reference can be used to create the final prediction block. For PUs in B slices, reference frames from both lists can be considered for prediction. In this case, either one reference is extracted from one of the lists to perform unidirectional prediction (from either List0 or List1), or two frames are extracted (one per list) to perform bi-directional prediction. When bi-directional prediction is applied, the final

prediction block uses two prediction sources, one from each reference picture list. The final prediction block is then given by the average of the two prediction sources. The concepts of uni- and bi-prediction are illustrated in 2.13

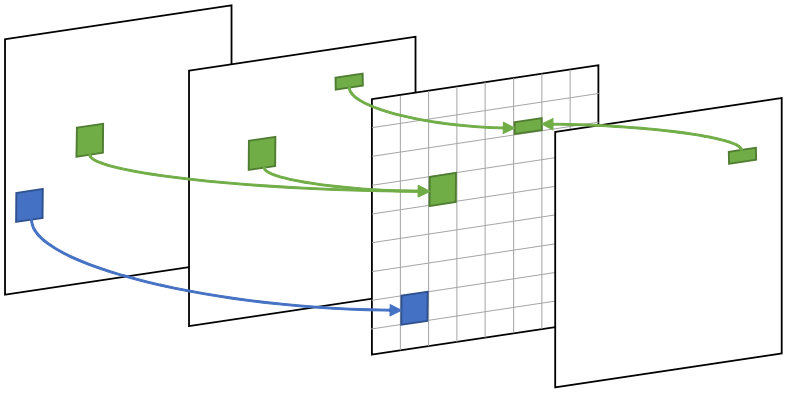


Figure 2.13: Uni-prediction (blue) and bi-prediction (green) in HEVC.

Reference lists are populated with previously encoded frames. Both lists have a limited number of available slots and typically List0 contains past reference frames (whose temporal index is smaller than the current index) while List1 contains reference frames from the future, in display order. However, reference frames from the past or future can be added to any of the two lists regardless of their temporal index.

It is important to note that, in contrast with Intra prediction, where mode signalling happens at the PU level but the prediction is computed at the TU level, in Inter prediction a single prediction is computed for each PU. In order to limit complexity, the usage of 4×4 Inter prediction blocks is forbidden in HEVC. This means that mode $N \times N$ cannot be used for Inter CUs of size 8×8. Moreover, for 4×8 and 8×4 PUs, bi-prediction is not allowed, meaning that only uni-directional prediction from List0 or from List1 is possible.

HEVC supports sub-pixel precision motion estimation up to quarter-precision accuracy [35]. This means that the samples of an area of a reference frame needed for Inter prediction are interpolated to create predictions with possibly higher precision. In HEVC, half-precision and quarter-precision samples can be computed separately [36].

Half-precision samples are computed using an 8-tap filter while quarter-precision samples are computed using one of two 7-tap filters. In the latter case, the interpolation filter to use is selected based on the location of the nearest integer-precision sample to the current quarter-precision sample being interpolated [37].

One important new feature introduced in HEVC with respect to AVC is the so-called merge mode predictions. The merge mode allows merging of motion information, i.e. sharing of identical motion vectors across a potentially large set of connected PUs. This provides the ability to encode homogeneous motion for arbitrarily shaped regions of a picture in a very efficient manner. In merge mode, the applicable motion information for the current PU is derived from a set of candidates. The list can contain up to 5 candidates, populated according to specific rules, and the selected candidate to use is indicated in the PU syntax by its index in the list (the merge index). The merge list can include multiple spatial candidates, as well as a possible temporal candidate. The spatial candidates considered for the merge list are illustrated in Figure 2.14.

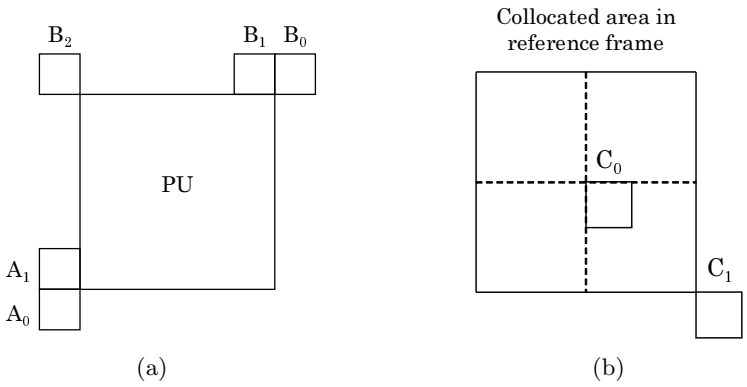


Figure 2.14: Candidate motion vectors considered in merge mode. a) Spatial candidates; b) Temporal candidates.

A maximum of 4 spatial candidates are included in the list. A temporal candidate is then considered, based on the location of the collocated area in a reference picture used for derivation of the temporal merge candidate. If the current PU is located at the lower boundary of the CTU or if the block at location C_0 is unavailable, the collocated PU is determined by the sample location C_1 in Figure 2.14(b). Otherwise, the collocated PU

is determined by the sample location C_0.

Potential candidate PUs which are outside of the current slice or Intra-coded are marked as unavailable and are ignored. If the number of entries in the list allows for further additions, combined bi-predictive merge candidates are added for B slices. If the size of the merge candidate list is still below the defined maximum, the merge candidate list is filled with zero motion vector merge candidates with increasing reference index for each empty position [38].

The merge mode prediction can also be used to skip the encoding of a given CU. At the very beginning of an Inter-coded CU, the syntax includes a binary flag indicating if skip mode is used in this CU or not. In case it is, the CU is predicted using a single PU of the same size (mode $2N \times 2N$). The only information following the skip flag is the index of the applicable merge candidate. No further information is coded for the CU, meaning that the motion vector derived from the merge prediction is applied without the addition of any residual information. One of the techniques proposed in Chapter 4 relies on the modification of the costs of merge mode candidates to prevent the appearance of contouring artefacts in compressed video.

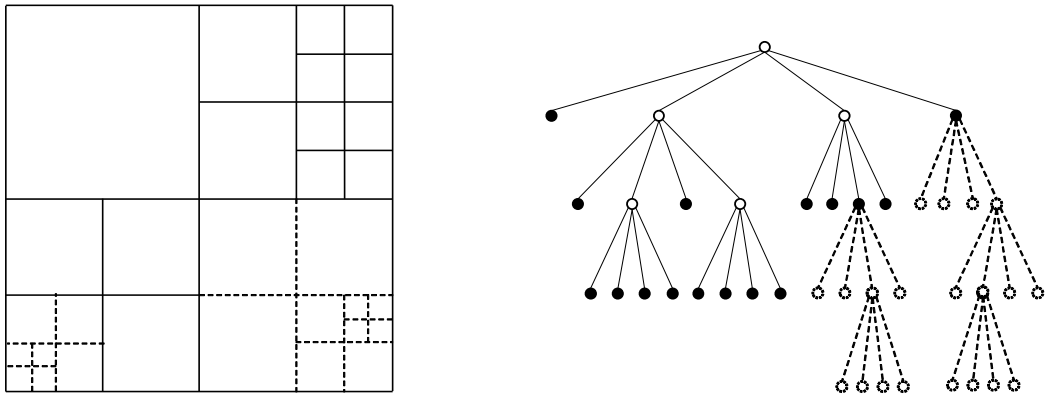
In case merge prediction is not used on a PU, the motion information is signalled in the bit stream, once for uni-directionally predicted PUs or twice for bi-directionally predicted PUs in B slices. Only the motion vector difference is encoded for each motion vector, computed as the difference between the selected motion vector and a motion vector used as prediction. This is derived using the so-called Advanced Motion Vector Prediction (AMVP) [39]. AMVP uses the same predictors considered for populating the merge candidate list shown in Figure 2.14. The motion vector prediction is selected for a PU as the element in the list that is more similar to the motion vector being predicted. A flag indicating which MV from the list is used as prediction is encoded in the motion information.

2.3.4 Residual coding

In case a CU is not skipped, the residual samples obtained using either Intra or Inter prediction are transformed and quantised. An additional partitioning level is supported in HEVC, enabling further partitioning of a CU into smaller Transform Units (TUs). Each TU is then transformed and quantised independently.

The partitioning of a CU into TUs is performed in accordance with the so-called Residual Quad-Tree (RQT) [40]. The partitioning approach is similar to the coding quad-tree structure used to split a CTU into smaller CUs. The root of the RQT is the CU, meaning that if no partitioning is applied, a TU of the same size of the CU is considered. For each CU, a transform split flag indicates if the transform is applied at this level or if the residual is split into four smaller TUs. In case of splitting, each resulting TU is assigned a split flag and may be similarly partitioned into four smaller TUs. This process is repeated until no further splitting is desirable or the minimum TU size is reached. A maximum RQT depth can also be specified, which also restricts the partitioning process. Similarly, a minimum and maximum TU size within the allowed range of 32×32 to 4×4 samples can be specified. Splitting is implicit when the CU size is larger than the maximum TU size (for example, for 64×64 CUs). Figure 2.15 illustrates the RQT partitioning into TUs [41].

For each TU, the transform and quantisation processes are specified using fixed-point integer operations. As previously mentioned, HEVC supports transform sizes of 4×4, 8×8, 16×16 and 32×32. These transforms are specified by the respective transform matrices, defined using integer approximations of the Discrete Co-sine Transform (DCT)-II matrix of the same size. The entries of the matrices of smaller sizes are all embedded in the 32×32 matrix, meaning that only this matrix is needed to derive all DCT transform matrices. An additional 4×4 Discrete Sine Transform (DST) matrix is also specified [42]. This transform is only applied to the residual of Intra predicted 4×4 blocks. The 4×4 DCT and DST matrices used in HEVC are the following:



(a) (b)

Figure 2.15: Residual quad-tree partitioning (dashed lines) on top of the corresponding coding quad-tree partitioning (solid lines).

$$T_{DCT4} = \begin{bmatrix} 64 & 64 & 64 & 64 \\ 83 & 36 & -36 & -83 \\ 64 & -64 & -64 & 64 \\ 36 & -83 & 83 & -36 \end{bmatrix}$$

$$T_{DST4} = \begin{bmatrix} 29 & 55 & 74 & 84 \\ 74 & 74 & 0 & -74 \\ 84 & -29 & -74 & 55 \\ 55 & -84 & 74 & -29 \end{bmatrix}$$

These matrices and the matrices of the remaining transform sizes are used to perform the transform process. Let $\mathbf{T}$ and $\mathbf{R}$ denote the transform matrix and a given residual block for a given TU, the block R is transformed into the transform coefficients C according to

$$\mathbf{C} = \mathbf{T} \cdot \mathbf{R} \cdot \mathbf{T}^{\mathsf{T}}. \tag{2.7}$$

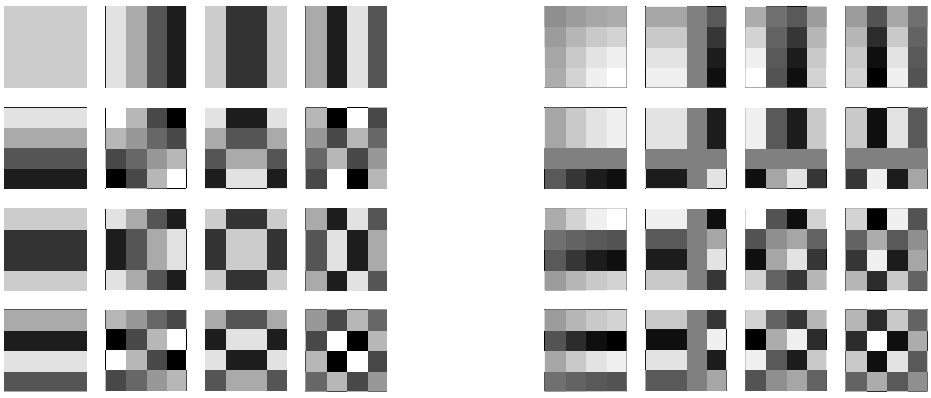
Since all transforms used in HEVC are orthogonal, the corresponding inverse trans-

form process is simply given by

$$\tilde{\mathbf{R}} = \mathbf{T}^{\intercal} \cdot \tilde{\mathbf{C}} \cdot \mathbf{T} \tag{2.8}$$

where $\tilde{\mathbf{C}}$ represents the transform coefficients after the quantisation and dequantisation processes and $\tilde{\mathbf{R}}$ represents the residual block after inverse transform.

The contribution of each transform coefficient of a transformed block can be visualised by plotting the set of DCT base functions. Figure 2.16 shows these base functions for the 4×4 DCT and DST transforms used in HEVC. Each base function is generated by applying the inverse transform operation to a block of transform coefficients with a single non-zero coefficient in the corresponding position. In the figure, the DC base function is located in the top-left corner. The horizontal and vertical frequencies increase to the right and to the bottom, respectively.



(a) (b)

Figure 2.16: a) DCT and b) DST base functions.

The DST used in HEVC is an integer implementation of the DST-VI. The DST was found to be more suitable for the spatial characteristics of the residual signal generated from Intra predictions [43]. However, the associated coding gains for larger blocks are less significant and therefore this type of transform is only applied to 4×4 blocks [44][45]. HEVC also introduces the transform skip mode where transform and inverse transform

operations are omitted, reducing possible ringing and blurring artefacts [46].

After the transform operation is applied, the resulting coefficients are quantised to a set of predefined values, according to a given quantisation step. The applicable quantisation step is indicated by the Quantisation Parameter (QP), which is an integer index ranging from 0 to 51. The QP value has a one-to-one mapping with the quantisation step used for quantisation.

Conceptually, for a given block of transform coefficients $\mathbf{C}$ with coefficients $c_{i,j}$, where i and j are the horizontal and vertical indices defining the position of the coefficient in the block, the scalar quantisation process in a typical HEVC encoder can be described by

$$v_{i,j} = \text{sgn}(c_{i,j}) \cdot \left\lfloor \frac{|c_{i,j}|}{\delta_q} + d \right\rfloor, \tag{2.9}$$

where $v_{i,j}$ denotes the resulting quantised coefficient level in position (i, j), δ_q denotes the quantisation step associated with the specified QP and d is a fixed offset used for rounding. The quantisation process is illustrated in Figure 2.17. In Figure 2.17, the values marked with a cross in the scale indicate the quantisation level that is obtained when a given transform coefficient falls inside the respective interval. The quantisation process is relevant to the estimation methods presented in Chapter 6 for encoding and uploading time control, as the statistics of the non zero quantised levels are used to provide accurate encoding time and bit rate estimations.

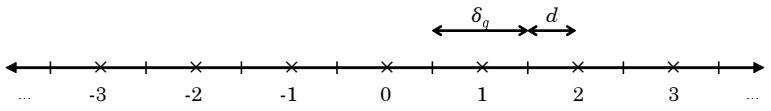


Figure 2.17: Illustration of the scalar quantisation process.

As previously mentioned, the quantisation process is where distortions are introduced during the encoding process, as transform coefficients cannot be fully recovered after it. At the decoder side, the inverse quantisation process is trivially achieved by scaling each

received quantised level with the same quantisation step used in the quantisation process:

$$\tilde{c}_{i,j} = \delta_q \cdot v_{i,j}. \tag{2.10}$$

In terms of the HEVC specification, the scaling process described by Eq. (2.10) is defined by equivalent integer scaling and shifting operations according to the used quantisation step δ_q [47]. Different scaling and shifting factors are defined according to the QP, slice type (Intra or Inter), colour component and transform size. These simplifications reduce the complexity associated with both software and hardware implementations of the standard. Similarly, the quantisation process conceptually described by Eq. (2.9) is expected to follow a similar approach, even though operations inherent to the encoding process are not specified by the standard. In fact, the techniques proposed in Chapters 3 and 4 rely on modifications of the quantisation process performed by the video encoder only, tuning it to provide better perceptual quality without sacrificing compliance with the HEVC standard.

After the quantisation process, all quantised coefficients are entropy encoded and written to the bit stream to be sent to the decoder [48].

2.3.5 Entropy coding

The entropy encoding of most syntax elements in HEVC is performed using Context-based Adaptive Binary Arithmetic Coding (CABAC) [49]. CABAC is already supported in the High profile of AVC as an alternative entropy encoding method to Context Adaptive Variable Length Coding (CAVLC) [50]. In HEVC, only CABAC has been adopted. The CABAC mechanism in HEVC is essentially the same as in AVC. Only some aspects of the design have been revised to achieve slightly higher compression efficiency and allow higher parallelism in the computations involved.

Arithmetic coding works by iteratively assigning intervals to symbols within a range

of real numbers delimited by two extremes a and b. One non-overlapping interval $[a_i b_i)$ is assigned to each symbol according to its probability of occurrence in such a way that the union of all intervals form the entire range, i.e. $[a_0, b_0) \cup [a_1, b_1) \cup ... [a_N, b_N) = [a, b)$. Initially, the range $a = 0$ and $b = 1$ is used, and at each encoded symbol i the range shortens to the interval $a = a_i$, $b = b_i$. After all symbols are coded, these are represented by any fractional number within the obtained range of values.

The arithmetic coder included in HEVC is binary. Fixed length, truncated unary or exp-golomb binarisation methods [49] are used, depending on the syntax element being entropy coded. Each binary value (bin) is then coded independently, meaning that only two ranges are assigned at each step. The context-adaptive property is related to the fact that the probabilities used to define the ranges during encoding are not fixed. They are assigned taking into account a set of available probability models that are shaped according to the statistics of previously coded symbols.

Overall, CABAC is a complex and powerful mechanism for entropy coding [51]. It is applied to the whole CTU syntax structure, leaving only some high-level syntax elements to be entropy encoded using fixed or variable length codes. More details about the operation of the CABAC engine, including adopted probability states, selection of contexts for each syntax element and details about encoding with context updates can be consulted in [49].

2.4 Conclusion

This chapter described the general background of the topics addressed in the following chapters of this thesis. Some basic image and video coding fundamentals were introduced, along with a general description of the typical hybrid block-based video coding approach. A more detailed description of the state-of-the art HEVC standard was also given, with emphasis on the tools that are most relevant to this thesis. It is important to refer, though, that there are other parts of the encoding process that were not described in

this chapter but also play an important role in the superior performance of HEVC with respect to previous standards. The Sample Adaptive Offset (SAO) [52] in-loop filter that was introduced in HEVC is one example (in addition to the already known deblocking filter from AVC [53]), along with tools that provide higher parallelisation options to the encoding and decoding processes, such as tiles [54] and Wavefront Parallel Processing (WPP) [55]. The details of these tools, as well as the details of HEVC's high level syntax [56], are not as relevant to this thesis as the aspects described in this chapter. For this reason, the reader is referred to the relevant literature for a detailed description of these tools [57][58].

Chapter 3

Perceptually-oriented HEVC encoding using just-noticeable distortion

As mentioned in the introduction of this report, the successful establishment of new video technologies, such as UHD TV and similar UHD video communication applications, is strongly dependent on the performance of the underlying video compression solutions. Even though the HEVC standard allows a significantly superior rate-distortion performance compared to previous video coding standards, further performance improvements are possible when exploiting the perceptual properties of the HVS.

In particular, the concept of Just Noticeable Distortion (JND) is based on the assumption that the HVS shows different sensitivities to different types of visual information. Image elements such as spatial frequency, pattern masking and luminance variations play an important role in the way images are perceived by the human brain. JND models aim at quantifying these differences and provide thresholds for image elements under which changes are not perceived by human viewers.

This chapter presents a novel perceptual-based solution, fully compliant with the HEVC standard, where a low complexity JND model is used to drive the encoder's Rate-Distortion Optimised Quantisation (RDOQ) process. By using a JND-model to modify the operation of RDOQ, the proposed technique provides an effective way to influence the decisions made at the encoder, based on the limitations of the HVS.

A brief review of the background work underpinning the proposed method is firstly given in Section 3.1. The detailed description of the proposed approach and the associated performance evaluation results are then presented in Sections 3.2 and 3.3, respectively. Finally, Section 3.4 closes this chapter with some final remarks.

3.1 Background work

JND models aim to define thresholds for image elements under which changes are not perceived by human viewers. Typically, two distinct approaches have been followed in the literature to model the perceptual limits of the HVS in terms of JND. The key difference between these approaches is the domain in which the JND thresholds are defined, which can be in the frequency domain or in the spatial domain. For this work, a frequency domain JND model was considered more suitable for integration into a state-of-the-art video coding solution, as the quantisation process is typically performed in the frequency domain (see Section 2.3.4).

The first advances made in exploiting the properties of the HVS using JND models were made for still images, where data from previous psychophysical experiments [59] were used to define a model for visibility thresholds when using DCT decomposition of images [60]. Later, Watson proposed the so-called DCTune model [61], where the model described in [60] was improved by considering image dependent parameters, notably considering luminance and contrast masking effects. These models aimed to specify perceptually optimised quantisation matrices for JPEG image compression and were used to specify different quantisation steps for different spatial frequencies in the DCT

domain. In the most recent HEVC standard, the usage of quantisation matrices is still supported. It is possible to set up the quantisation process to use default or user-defined quantisation matrices. The usage of quantisation matrices must be signalled in the bit stream to ensure that the equivalent inverse quantisation process is applied at the decoder.

In 2003, Zhang et. al. [62] used a threshold profile similar to the one in [60] and proposed an improved contrast masking factor based on a block classification technique. This classification was based on the magnitude of the DCT coefficients considered to represent low, medium and high spatial frequencies. This classification was proposed for a fixed transform size of 8×8, in order to be easily adapted to JPEG image compression. Later, Jia et. al. [63] proposed a JND model for video where the proposed base threshold was defined by taking into account the retinal image velocity and the amount of motion in the video. The luminance adaption and contrast masking factors used in this model were identical to the ones in [62].

In 2005, Yang et. al. [64] proposed a method for pre-processing prediction residuals based on a pixel domain JND model introduced in [65]. This pixel domain JND model was used to reduce the prediction residual prior to the transform operation so that only the perceptually relevant residual would go into the transform and quantisation processes. This method was developed for the MPEG-2 TM5 encoder. In 2009, Mak et. al. [66] proposed a similar suppression approach to the one in [64], but based on a transform domain JND model. The technique consisted of discarding the residual coefficients whose absolute values were lower than the JND thresholds. This technique was integrated into an AVC encoder.

Later, Chen et. al. [67] proposed a method for macroblock quantisation adjustment in AVC based on the pixel domain JND model in [65]. This JND model was combined with a foveation model to take into account both threshold visibility and visual eccentricity. The method was used at the macroblock level to select the optimal QP and Lagrangean multiplier in the RDO process according to the model.

In 2011, Naccari et. al. [68] proposed an AVC-based perceptual video codec that adaptively selected, at the encoder, the quantisation step of each transform coefficient. This adaptive selection was based on the JND model defined in [69]. At the decoder, a method was proposed to predict the right quantisation step to use for inverse quantisation of each coefficient, to avoid additional signalling bit rate. Due to the required adaptation in the decoder operation, this technique is not compliant with the AVC standard. This technique was further extended to an HEVC video codec in [70]. Later, a new perceptual video coding tool was proposed to adjust the quantisation step of each transform coefficient based on the HVS luminance masking effects [71]. The technique was designed for an efficient transmission of the additional luminance masking parameters and low-complexity implementation.

More recently, in 2015, Kim et. al. [72] proposed a solution fully compliant with the HEVC standard where the model in [69] was adjusted to cope with the different transform sizes used in HEVC. The modified JND model is then used to lower and suppress the values of the transform coefficients before quantisation. Similarly to the technique in [61], reducing the residual before the quantisation process does not account for the additional error introduced by the quantisation process, which may introduce further distortions above the JND threshold. An average bit rate reduction of around 16% with negligible subjective quality loss was reported.

In this chapter, an alternative low-complexity JND-driven solution is proposed. The proposed method is fully compliant with the HEVC standard and it targets to modify the decisions made at the encoder according to the perceptual properties of the HVS. It is important to note that using the quantisation matrices supported by HEVC does not provide a content dependent solution, as the matrices are fixed at the frame level. Moreover, to be able to tune the quantisation step at the transform coefficient level, non-normative approaches need to be adopted, as the standard does not provide any mechanism for this purpose.

By perceptually modifying the operation of the RDOQ process, the proposed approach

is capable of significantly reducing the bit rate associated with the encoded bit stream and preserving the output video quality, as shown in the rest of this chapter. The complexity introduced by the proposed technique is very low, making it particularly suitable for practical encoding applications.

3.2 Rate-distortion optimised quantisation using just noticeable distortion

The proposed mechanism adopts a JND model to modify the choices made at the encoder according to the limits of human visual perception. A simplified diagram of the workflow adopted by the proposed mechanism is depicted in Figure 3.1.

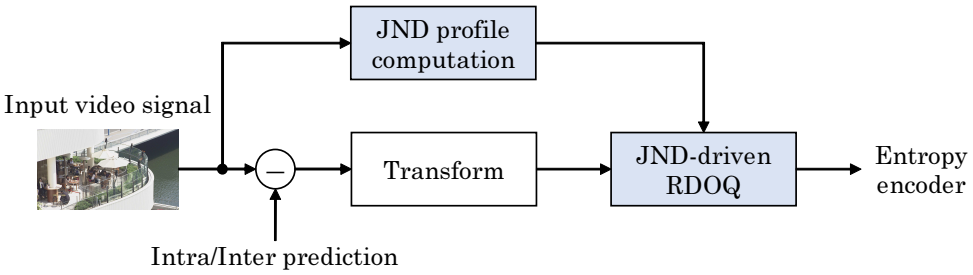


Figure 3.1: High-level diagram of the workflow adopted by the proposed approach.

As shown in Figure 3.1, frames from the input video signal are used to compute the JND profile. This JND profile defines a set of thresholds that identify the maximum distortion that can be introduced to the video signal without being perceived by the HVS. This profile is then used to drive the quantisation process, by modifying the costs used in the RDOQ process performed on the transform coefficients of the residual signal. The selected quantised coefficient levels are then entropy encoded to be included in the compressed bit stream.

The JND profile computation and its integration in the RDOQ process of a video encoder are the main modifications introduced by the proposed mechanism to the typical

video encoding workflow. These two processes are described in detail in the following subsections.

3.2.1 JND profile computation

The adoption of a low-complexity model is essential to enable the proposed solution to be used in practical video compression applications. For complexity reduction purposes, the model in [69] was selected and adapted to the different transform sizes allowed in HEVC, using the method based on the spatial summation effect in [73]. It is important to note that, since the proposed integration technique of the JND model into an HEVC encoder is model-independent, the selected model can be replaced by a more accurate and sophisticated model depending on the complexity restrictions of the target application.

For a given transform block b, the adopted JND threshold, $T_{JND}(b,i,j)$, associated with the transform coefficient with indices (i,j) is computed as

$$T_{JND}(b,i,j) = T_B(b,i,j) \cdot F_{LM}(b) \cdot F_{CM}(b,i,j). \tag{3.1}$$

As seen in Eq. (3.1), the JND threshold $T_{JND}(b,i,j)$ is given by the product of a base threshold $T_B(b,i,j)$, a luminance masking factor $F_{LM}(b)$ and a contrast masking factor $F_{CM}(b,i,j)$. The following subsections briefly describe each of these components of the adopted JND model.

3.2.1.1 Base threshold

The base threshold accounts for the different sensitivity of the HVS to distortions added to different spatial frequencies. For a given transform block, b, of size $N \times N$, $T_B(b,i,j)$ is given by

$$T_B(b,i,j) = S(N) \cdot \frac{1}{\phi_i \phi_j} \cdot \frac{H(f_{i,j})^{-1}}{r + (1-r) \cdot \cos^2 \varphi_{i,j}}, \tag{3.2}$$

where $H(f_{i,j})$ is the Contrast Sensitivity Function (CSF), $S(N)$ is the spatial summation effect, ϕ_i and ϕ_j are the DCT normalisation factors and the term r + $(1-r) \cdot \cos^2 \varphi_{i,j}$ accounts for the different sensitivity of the HVS regarding directionality. All parameters in Eq. (3.2) were computed as in [69], with the exception of the CSF and $S(N)$. The adopted CSF is given by

$$H(f_{i,j}) = \left(1 - a + \frac{f_{i,j}}{f_0}\right) \cdot e^{-\left(\frac{f_{i,j}}{f_0}\right)^p}, \tag{3.3}$$

where $f_{i,j}$ represents the spatial frequency, computed as in [69], and $f_0 = 1.7377$, $a = 1.0465$ and $p = 0.6937$ are the best fitting parameters to a CSF of this type, according to the experiments conducted in [74] for a dataset of 43 image patterns. The parameters used in [69] were not considered in this case since they were empirically estimated based on a fixed transform size experiment (8×8). Instead, the $S(N)$ factor compensates for spatial summation, which accounts for the effect of having simultaneous distortions over a range of spatial frequencies in a given frame area. Similarly to [73], the spatial summation effect was modelled as

$$S(N) = N^{\left(-\frac{2}{\tau}\right)} \tag{3.4}$$

in order to adapt the base threshold to the transform size used. In Eq. (3.4), the parameter τ was set to 1.873 according to the experiments conducted in [73].

3.2.1.2 Luminance adaptation factor

The luminance adaptation factor accounts for the fact that visibility thresholds depend on the average brightness level of a given block. The HVS is less sensitive to changes in brighter and darker backgrounds and therefore the visibility threshold in these conditions

can be increased.

As in [69], for a given transform block b, the luminance adaptation factor is given by

$$F_{lum}(b) = \begin{cases} \frac{(60-\bar{I})}{150} + 1, & \bar{I} \leq 60 \\ 1, & 60 < \bar{I} < 170 \\ \frac{(\bar{I}-170)}{425} + 1, & \bar{I} \geq 170 \end{cases} \tag{3.5}$$

where $\bar{I}$ denotes the average luma intensity value of the samples inside block b. Considering a bit depth of 8 bits per sample, the graphical representation of the luminance adaptation factor for each average luma intensity value is shown in Figure 3.2.

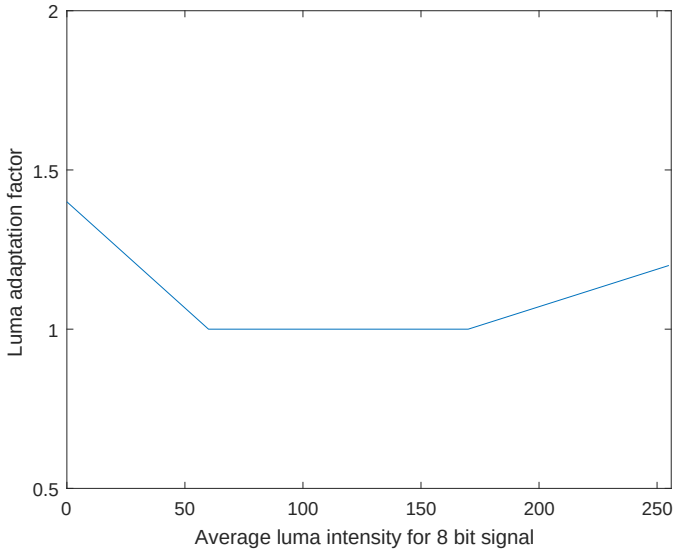


Figure 3.2: Luminance adaptation factor according to the average luma intensity value in an image block.

Figure 3.3 shows two examples of the luminance adaptation factor obtained for a given frame for two different transform sizes.

(a) (b) (c)

Figure 3.3: Luminance adaptation factor obtained for a given frame. a) Original frame; b) 4×4 luminance adaptation map; c) 32×32 luminance adaptation map.

3.2.1.3 Contrast masking factor

The contrast masking factor accounts for the reduction of visual sensitivity in one visual component in the presence of another. Typically, distortions are more difficult to notice when introduced in areas where texture energy is high. Therefore, a contrast masking factor is used to elevate the threshold of each coefficient in a given block depending on the texture characteristics of the visual content in this area.

For the purpose of computing $F_{CM}(b,i,j)$, the Canny edge detector [75] is first applied to the whole frame and for a given DCT transform block size $N \times N$, each block is classified as a "Plane", "Edge" or "Texture" block according to

$$\text{Block type} = \begin{cases} \text{Plane}, & \rho_{\text{edge}} \leq \alpha \\ \text{Edge}, & \alpha < \rho_{\text{edge}} \leq \beta \\ \text{Texture}, & \rho_{\text{edge}} \geq \beta \end{cases}, \tag{3.6}$$

where α and β are empirically set to 0.1 and 0.2, respectively, and ρ_{edge} is the density of edge pixels inside the block identified by the Canny edge operator. Formally, ρ_{edge} is given by

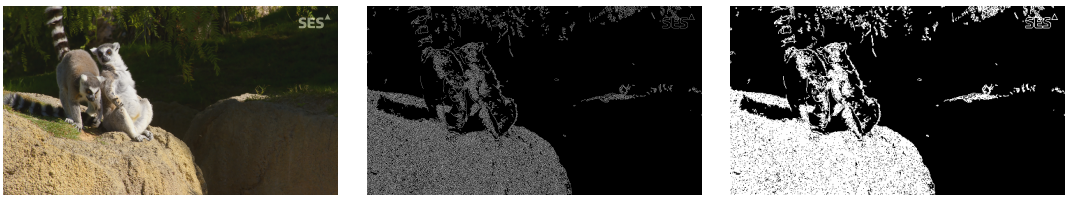
$$\rho_{\text{edge}} = \frac{P_{\text{edge}}}{N \times N} \tag{3.7}$$

where P_{edge} is the number of edge pixels inside the block.

For a given coefficient with indexes i and j inside block b, the final elevation factor is given by

$$F_{CM}(b,i,j) = \begin{cases} 1, & \text{Plane or Edge} \\ 2.25, & \left(i^2+j^2\right) \leq 2N \text{ in Texture} \cdot \\ 1.25, & \left(i^2+j^2\right) > 2N \text{ in Texture} \end{cases} \tag{3.8}$$

Contrarily to the contrast masking factor in [69], the term introduced following the Foley-Boynton method [76] was not considered in the proposed approach. This was due to the required computation of the transform coefficients of the original frame, increasing this way the complexity of the overall solution. Figure 3.4 shows the result of the Canny edge detection for an example frame and the respective contrast masking factor map obtained for a transform size of 4×4.



(a) (b) (c)

Figure 3.4: Contrast masking factor map generation. a) Original frame; b) Output of the Canny edge detection; c) 4×4 contrast masking factor map.

3.2.2 JND-driven rate-distortion optimised quantisation

The method to integrate the selected JND model into the reference HEVC encoder consists of modifying the RDOQ process in an HEVC encoder according to the thresholds defined by the JND model described in the previous section. In this subsection, a brief description of the RDOQ process is first given, followed by the description of the proposed modifications to turn it into a perceptually tuned quantisation tool.

3.2.2.1 Rate-distortion optimised quantisation

The RDOQ process [77] consists of optimising the choice of each coefficient level obtained after quantising a given transform coefficient, considering both the introduced distortion and the associated bit rate. When the RDOQ tool is not used, the nearest integer rounding rule is used by the reference HEVC encoder to round a given quantised coefficient to the nearest integer level, L. Even though this rounding process minimises the distortion introduced by quantisation, choosing a different quantised level may be beneficial when considering also the associated bit rate. Therefore, when RDOQ is enabled in recent versions of the HEVC reference software, the levels L, $L-1$ and 0 are also evaluated. The coefficient level that shows the lowest rate-distortion cost is selected. Figure 3.5 shows an example of the reconstructed values corresponding to the levels tested by the RDOQ process.

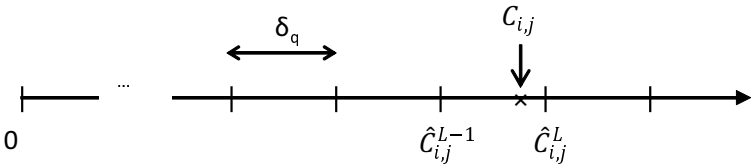


Figure 3.5: Candidates tested when using the RDOQ process to quantise a given transform coefficient $C_{i,j}$.

This optimisation is performed for each non-zero coefficient level in a quantised TB. The cost of each coefficient level tested by the RDOQ process, J, is computed according to

$$J = D_x + \lambda \cdot R_x \tag{3.9}$$

where D_x is the distortion introduced by the selection of a given candidate level x (i.e, L, $L-1$ or 0), λ is the Lagrangean multiplier and R_x is the bit rate associated with each level being tested. In Eq. (3.9), the distortion, D_x, is the square of the error, E_x, introduced by the quantisation process, given by

$$E_x = \left| C_{i,j} - \tilde{C}^x_{i,j} \right|. \tag{3.10}$$

It is important to recall that the HEVC standard only specifies the syntax of the encoded bit stream and the decoding process. Thus, adjusting the quantised levels to minimise the RD cost is a decision made at the encoder and therefore any rule for selecting the quantised levels can be applied for this purpose without sacrificing compliance with the standard.

3.2.2.2 Modified RDOQ based on JND

As described in Section 3.2.1, the adopted JND profile defines a threshold for each transform coefficient that represents the maximum amount of distortion that can be added to that coefficient without causing a perception of distortion by the HVS. It is therefore possible to modify the value of D_x according to this threshold in order to take into consideration the limitations of the HVS when computing the cost of each optimised level being tested.

Assuming that $T_{JND}(b,i,j)$ denotes the visibility threshold of the coefficient in position (i,j) of a given transform block b, the proposed modified distortion, D'_x, to be used in the cost computation of each candidate coefficient level, is computed based on a different error, E'_x, given by

$$E'_x = \begin{cases} 0, & E_x \leq T_{JND}(b,i,j) \\ E_x - T_{JND}(b,i,j), & E_x > T_{JND}(b,i,j) \end{cases} \tag{3.11}$$

In practice, replacing D_x for D'_x in the cost computation means that any distortion lower than that allowed by the JND threshold should be considered null, as this distortion is not perceptually noticeable by the HVS. In case this distortion is higher than the

threshold, only the difference between these two values should be considered in the RDOQ cost computation.

Finally, it is important to note that the base threshold component of the selected JND model, $T_B(b, i, j)$, is content independent. This means that this threshold only needs to be computed once for the each possible transform size used by the encoder. On the other hand, the luminance adaptation factor and contrast masking factor depend are content dependent and therefore have to be computed and stored in memory for each frame and transform size. This allows the relevant thresholds for each TB size and position in a frame to be available to use when needed during the RDOQ process.

3.3 Performance evaluation

Experiments were conducted to assess the bit rate reduction capabilities of the proposed solution. The experiments were performed for the first 100 frames of 3 UHD test sequences and 3 HD test sequences, under the Random Access test configuration [78], using the HEVC reference software HM 16.2 for four different QPs. The results of the proposed technique implemented on top of the reference software were compared with the reference software. In both cases, RDOQ was enabled. For all obtained results, shown in Table 3-A, the decoded sequences were evaluated and no visual quality degradation was observed with respect to the decoded output of the HEVC reference software, despite the small PSNR losses.

As shown in Table 3-A, the proposed JND-driven RDOQ technique is able to significantly reduce the bit rate for lower QPs in all sequences, especially for the 3 UHD sequences tested, where this reduction can go up to 62%. Higher reductions are expected in lower QPs since lower quantisation steps increase the number of cases where the quantisation error is lower than the JND threshold.

As expected, a small loss in terms of PSNR is introduced when using the pro-

Table 3-A: Performance evaluation of the proposed JND-driven RDOQ.

		HM-RDOQ		JND-RDOQ				
Sequence	QP	Rate [kb/s]	Y PSNR [dB]	Rate [kb/s]	Y PSNR [dB]	Rate saving	PSNR diff. [dB]	Enc. time diff.
ShowDrummer 3840x2160 60 Hz	22	62823	38.27	26614	37.86	-58%	-0.41	-1%
	27	8224	37.52	7446	37.47	-9%	-0.05	9%
	32	3827	36.92	3767	36.89	-2%	-0.03	10%
	37	2098	36.00	2082	35.98	-1%	-0.02	10%
HomelessSleeping 3840x2160 60 Hz	22	85844	37.38	32302	36.82	-62%	-0.56	-4%
	27	8276	36.52	6277	36.48	-24%	-0.04	8%
	32	2810	36.06	2743	36.04	-2%	-0.02	9%
	37	1393	35.41	1380	35.40	-1%	-0.01	10%
YoungDancers1 3840x2160 50 Hz	22	61201	40.38	30206	39.22	-51%	-1.16	0%
	27	10726	38.74	7086	38.55	-34%	-0.20	7%
	32	3021	38.05	2821	38.02	-7%	-0.04	9%
	37	1623	37.29	1615	37.26	0%	-0.03	9%
BasketballDrive 1920x1080 50 Hz	22	17254	39.30	13502	38.93	-22%	-0.36	4%
	27	6071	37.70	5740	37.57	-5%	-0.13	11%
	32	2884	35.92	2829	35.84	-2%	-0.07	12%
	37	1537	33.97	1522	33.94	-1%	-0.03	11%
BQTerrace 1920x1080 60 Hz	22	39832	37.99	26556	36.94	-33%	-1.05	2%
	27	10001	35.54	8489	35.32	-15%	-0.22	9%
	32	3654	33.79	3491	33.69	-4%	-0.11	11%
	37	1672	31.76	1650	31.71	-1%	-0.05	11%
Cactus 1920x1080 50 Hz	22	20816	38.43	15924	37.96	-24%	-0.47	6%
	27	6791	36.75	6363	36.57	-6%	-0.19	11%
	32	3230	34.84	3159	34.74	-2%	-0.09	13%
	37	1675	32.65	1654	32.60	-1%	-0.05	13%

posed JND-driven RDOQ solution. Nonetheless, as previously mentioned, all decoded sequences were visually inspected and no visual quality degradations were identified. Since the main target of the JND-driven RDOQ technique is to perceptually optimise the performance of the RDOQ decisions in an HEVC encoder, the PSNR loss is not as relevant as the subjective output video quality of the decoded sequences.

For higher quality test points, the extra complexity introduced by the proposed technique is compensated by a reduction in the number of non-zero coefficients to encode, leading to even lower overall encoding times in the case of UHD sequences. For the remaining QPs, the overall additional complexity introduced for all sequences by the

proposed technique is in general low (average encoding time increase of 8%).

From the results in Table 3-A, it is clear that the proposed JND-driven RDOQ solution shows higher bit rate reduction capabilities when the target qualities are high. The solution is able to reduce the bit rates by reducing the amount of perceptually irrelevant visual information in the decoded sequences, providing the same output perceptual quality for significantly lower bit rate.

To further evaluate the performance of the proposed solution for higher qualities, an alternative perceptual quality metric was also used to evaluate the quality of the decoded sequences, in an attempt to have a more perceptually oriented evaluation. The selected metric to additionally evaluate the quality of the decoded sequences was the Video Quality Metric (VQM) [16], briefly described in Chapter 2, which shows a better correlation with Mean Opinion Score (MOS) tests than PSNR, according to [16]. In contrast with PSNR, the lower the VQM value, the higher the quality of the sequence being evaluated. The results obtained are shown in Tables 3-B and 3-C.

Table 3-B: JND-driven RDOQ performance analysis for lower QPs using VQM.

	HM-RDOQ			JND-RDOQ		
	QP	Rate [kb/s]	VQM	QP	Rate [kb/s]	VQM
HomelessSleeping	26	13743	0.0436	25	10976	0.0417
ShowDrummer	24	28920	0.9841	22	26614	0.9797
YoungDancers1	22	61201	1.1772	20	54485	1.1764

Table 3-C: JND-driven RDOQ performance analysis for lower QPs using VQM (bit rate savings and output quality differences).

	Bit rate saving	PSNR diff. [dB]	VQM diff.
HomelessSleeping	-20%	-0.18	-0.0019
ShowDrummer	-8%	-0.02	-0.0044
YoungDancers1	-11%	-0.72	-0.0007

Similarly to the previous results presented in this section, negative values in the bit rate saving column represent bit rate reductions achieved by the proposed JND-driven RDOQ technique with respect to the HEVC reference software. In the VQM difference

column, negative values represent an increase of output video quality according to the VQM metric and negative values in the PSNR difference column represent a quality decrease in terms of PSNR.

From the VQM results in Tables 3-B and 3-C, it is possible to conclude that, for these specific target qualities, the proposed JND-driven RDOQ technique is able to increase the quality of the decoded sequences and, at the same time, reduce the bit rate by up to 20% for the UHD sequences tested.

3.4 Conclusion

This chapter presented a novel technique for integrating a JND model into an HEVC encoder, allowing a perceptually-oriented selection of the quantised levels by the RDOQ process. This technique modifies the decisions made at the encoder side, meaning that a fully compliant bit stream is generated with the proposed solution. The results obtained show significant bit rate reductions with respect to the HEVC reference software, for the same perceived output visual quality, especially for UHD video content. Higher bit rate reductions are achieved for higher target qualities, as more distortions can be introduced without being perceptually noticed in these cases. In some cases, small video quality improvements can even be achieved with lower bit rates, when considering a perceptually-oriented video quality metric to evaluate the reconstructed quality. Finally, the required extra complexity is very low, making this technique suitable for integration into any HEVC encoder that can use RDOQ.

Chapter 4

Contouring artefacts prevention in HEVC encoded video

Following the perceptual optimisation technique proposed in the previous chapter, this chapter focuses on reducing the visibility of compression artefacts that may damage the perceptual quality of the encoded video. It is essential to guarantee that the perceptual quality of the compressed video is not damaged by possible compression artefacts that might deteriorate the superior quality of experience that, for example, UHD video formats are able to provide. In this context, it is relevant to investigate some particular contouring artefacts, also referred to as banding, that appear in flat backgrounds of some compressed UHD video sequences. These artefacts contribute to an undesirable degradation of the decoded output video quality. For some UHD video sequences, this degradation occurs also when light compression is applied, even when the objective quality measured in terms of PSNR is very high (e.g. around 45 dB).

Contouring artefacts tend to appear in spatially slowly varying flat backgrounds, usually associated with smooth spatial light variations. These light variations are represented by small local pixel variations in the original video signal. During the compression process, the coarse quantisation of the residuals in the frequency domain leads to a loss

of the small pixel variations responsible for smooth spatial transitions.

Figure 4.1 shows a visual example that highlights the effects of quantisation in the appearance of contouring artefacts. The selected part of the original frame is shown when different QPs are used for compression. The luma intensity values in the examples in Figure 4.1 were scaled for a better perception of the effects of contouring artefacts in the visual quality of the decoded video sequences.

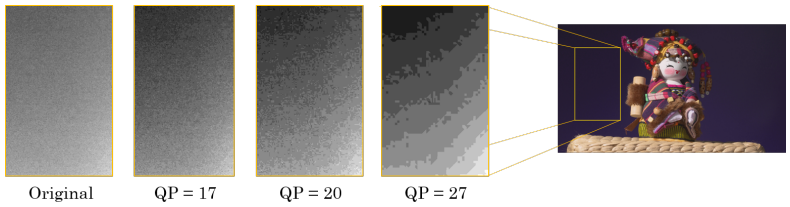


Figure 4.1: Contouring artefacts caused by quantisation in compressed video sequences. The same area of an original frame of the *Ningyo* sequence is shown when different QPs are used in the video compression process.

The RDO process performed by typical hybrid block-based video encoders controls the overall encoding process, assuring the best trade-off between output decoded quality, measured according to PSNR, and associated output bit rate. Since the distortion in this mechanism is controlled by the MSE, the loss of the small detail in flat background regions is not reflected in the output decoded quality and the decoded content in these areas tends to be averaged, without significant penalty in the final PSNR value. However, this loss of detail turns smooth variations into perceptually annoying layers in smooth backgrounds that significantly degrade the perceived quality of the decoded video.

This chapter proposes two methods based on the reduction of the QP values in areas prone to contouring artefacts to prevent them from appearing in compressed UHD sequences. The first method uses a detection technique previously proposed in the literature to identify areas of the video prone to contouring artefacts and increases the video quality in these areas by performing quantisation with lower QPs. The second method is an extension of this technique based on additional RD cost modifications in

contouring areas, aiming to further prevent the visibility of contouring artefacts in the decoded video sequences. The latter extension is also able to slightly reduce the extra bit rate associated with the first technique for some video sequences.

A brief overview of some relevant solutions previously proposed in the literature to avoid contouring artefacts is given in Section 4.1. Section 4.2 then describes the two proposed contouring prevention techniques, including including the implementation details associated with reducing the QP in contouring areas and the associated challenges. Finally, Section 4.3 reports the performance evaluation of the techniques proposed in this chapter.

4.1 Background work

Several solutions have been proposed in the literature to decrease the visibility of contouring artefacts in decoded video sequences. These solutions can typically be grouped into the two main approaches described in the next paragraphs. In short, the first one relies on image/video post-processing techniques, meaning that the encoding/decoding process is not changed. The second one involves changes in the behaviour of the video encoding/decoding process to achieve a similar goal.

Most of the solutions proposed to prevent contouring artefacts use post-processing techniques at the decoder side after decoding the received video sequences, without interfering in the encoding/decoding process itself. With such an approach, each decoder needs to apply its own post processing algorithm, meaning that the decoded output may vary from decoder to decoder and increased decoder complexity is needed. In [79], Ahn et. al. proposed a method to detect flat areas that might suffer from contouring artefacts based on local pixel density and standard deviation. For the flat regions identified, a random shuffler is applied to shuffle pixel positions in a block-based fashion, followed by a low pass filter. Finally, dithering is applied using an error diffusion filter to mask the contouring effects. Several other dithering techniques were also proposed to reduce

contouring artefacts, such as the one in [80], where contouring artefacts suppression is achieved by a dithering algorithm based on multi-scale probabilistic analysis on the neighbourhood of each pixel. More recently, Wang et. al. [81] proposed another low complexity block-based dithering method to recover gradient and boundary smoothness after compression. This dithering technique was applied only in image blocks prone to suffer from contouring artefacts, identified in the original frames by a contouring detection method based on the average pixel value level difference between a given block and its 8 neighbouring blocks. Finally, Lee et. al. [82] proposed a method to reduce the visibility of contouring artefacts by means of variable-size directional filtering applied orthogonally to the direction of the detected contouring artefacts.

Another approach to prevent contouring artefacts from appearing in the decoded video sequences is to modify the behaviour of the encoding/decoding process. With such an approach, the output decoded video is not dependent on any post-processing algorithm implemented at the decoder and therefore higher uniformity is achieved in terms of the decoded video content each decoder outputs to the final user. Yoo et. al. [83] proposed a method to reduce the visibility of contouring artefacts by injecting in-loop pseudo-random noise after the in-loop deblocking filter of an AVC encoder. Visual quality improvements are reported with low rate-distortion losses. Tan et. al. [84] also addressed the problem of contouring artefacts during the HEVC standardisation process. The problem was considered to stem from a discontinuity in the reference samples located at 32×32 block boundaries used to generate Intra prediction samples. This discontinuity was then propagated to the remaining frames, causing the undesirable contouring artefacts. The proposed solution consisted of applying a bi-linear interpolation of the reference samples using the corner samples only in 32×32 blocks, considered to be the ones where the effect was more visible. More recently, Casali et al. [85] proposed an HEVC compliant method to adjust the output of the quantisation process of an HEVC encoder by managing the size of the quantisation dead-zone. Successful contouring artefacts removal was reported for HEVC Intra coding with low RD performance

losses.

In this chapter, two techniques are presented to prevent contouring artefacts from appearing in compressed UHD sequences, both based on using finer quantisation in a fully HEVC compliant video compression solution. Therefore, the described techniques fall into the second approach described in this section since no post processing technique is considered at the decoder.

4.2 Region-adaptive quantisation for contouring artefacts prevention

Both techniques proposed and analysed in this section rely on reducing the QP used by a video encoder in order to enhance the perceived quality of the areas of the video sequences that might suffer from contouring artefacts. Figure 4.2 illustrates the workflow of the relevant parts of a typical video encoder where this approach is applied.

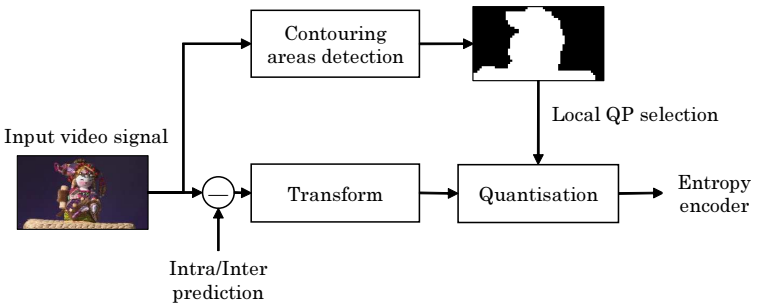


Figure 4.2: Workflow of the QP reduction approach integrated into a typical HEVC encoder architecture.

As depicted in Figure 4.2, the contouring areas detection process takes as input the original video signal and outputs a binary map indicating which areas of the video frame are prone to suffer from contouring artefacts after compression and which areas are not. This classification was assured by the block-based method proposed in [81], which is able to perform the decision at an arbitrary $N \times N$ block level. Figure 4.3 shows two examples of the contouring maps generated for a given frame of two different sequences

using $N = 64$. This block size was selected as it proved to be the most adequate in terms of the accuracy of the generated contouring maps for the video sequences tested in this work.

The black areas in Figures 4.3(b) and 4.3(d) represent areas where contouring may occur after compression (contouring areas), whereas white areas represent areas where contouring is unlikely to occur (non-contouring areas). The generated contouring map is then used to command the quantisation parameter used in the quantisation process of an HEVC encoder. This way, for areas where the compression process is likely to generate contouring artefacts, the QP used is lower in order to preserve the details in the prediction residual. For the remaining areas, the base QP set according to the desired level of compression is used.

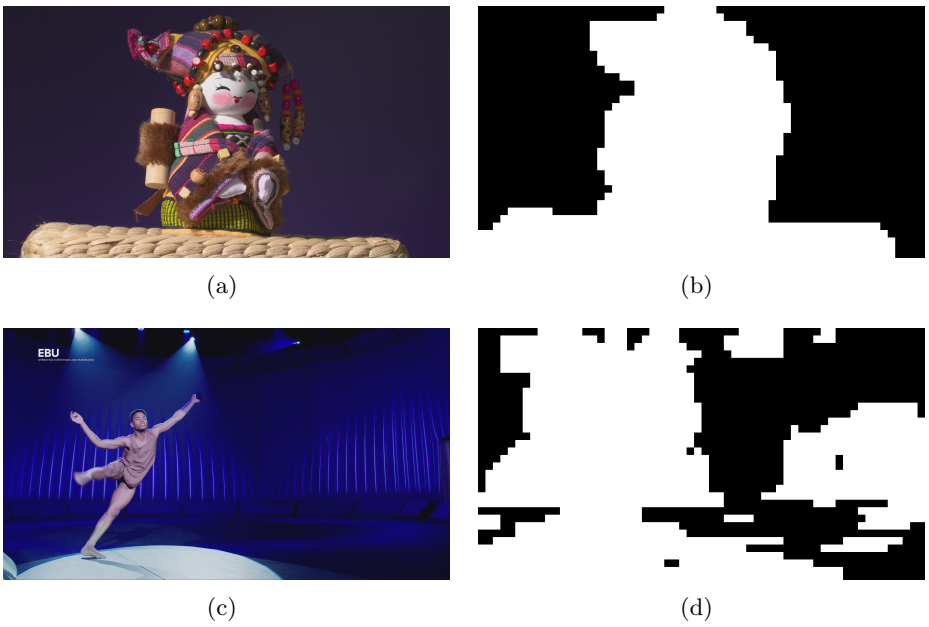


Figure 4.3: Original sample frames of the test sequences a) Ningyo and c) YoungDancers. Corresponding contouring maps with a resolution of 64×64 blocks for b) Ningyo and d) YoungDancers.

Subsection 4.2.1 describes in detail the proposed region-adaptive fine quantisation technique based on the described QP reduction approach, focusing on the implementation aspects and associated challenges. Subsection 4.2.2 describes the proposed extension to

improve the latter technique using modified rate-distortion costs.

4.2.1 Fine quantisation in contouring areas

The contouring map generated by the contouring areas detection process is used to select the QP applied in the quantisation process. Video coding solutions typically have mechanisms to support such local QP adaptation. For example, in this work, the QP selection is performed at the CTU level. It is important to note, however, that the HEVC syntax allows the usage of different QPs at the CU level. The selection of the QP to be used in each CTU is performed at the encoder and signalled to the decoder through the Delta QP syntax element. Therefore, the proposed solution generates a bit stream in conformance with the HEVC standard and no changes are required at the decoder side.

When applying the QP reduction under RA encoding conditions [78], many factors need to be taken into account to prevent contouring artefacts from appearing in the decoded sequences. For Intra frames, contouring artefacts are mainly caused by coarse quantisation of the prediction residual. In smooth background areas, the selected Intra predictions are typically flat (especially when DC Intra prediction mode is used) and after quantising the resulting prediction residual with high quantisation steps, the very small variations in the original pixel values become totally flat in the reconstructed image. As spatial light variations occur smoothly in the background, flat areas start to appear as layers in the reconstructed pictures, creating the undesirable and perceptually annoying contouring artefacts.

The causes of the appearance of contouring artefacts in Inter frames, where motion compensation prediction is used, are similar to the causes observed for the Intra case. However, for Inter frames, there is the possibility of choosing good predictions that preserve these small variations in the original images, if the previously encoded frames used as reference are already contouring-free. Considering the proposed method of reducing the QP in contouring-prone areas, a deeper analysis of what is observed in Inter frames

for lower and higher target qualities is given in the following:

1. **Lower base QPs** - The proposed QP reduction technique forces the RDO algorithm in an HEVC encoder to use a lower QP in areas where contouring artefacts are likely to appear. This method aims to preserve some noise in smooth areas, which are more vulnerable to contouring artefacts. However, for lower base QPs (e.g. QP = 22), this strategy is affected by the higher Lagrangean multiplier (λ) used for Inter frames. The Lagrangean multiplier controls the RDO process and it is commonly assigned at the frame level, according to the type of frame (Intra or Inter), QP and relative position of the frame inside the Structure Of Pictures (SOP). As λ is assigned with a higher value in Inter frames, the RDO process selects more often the coding modes and motion vectors that lead to lower bit rates. This means that setting the residual to zero will be, in most cases, the preferred option in contouring areas, even if the QP is significantly low. As the quality of the prediction signal is measured according to the distortion between original and predicted signals, flat areas are usually selected as prediction references, as these often show a lower sum of absolute differences. For this reason, the absence of residual makes flat areas appear as layers in the reconstructed pictures, causing slightly visible contouring artefacts. It is important to note though that the contouring artefacts are much less visible with the proposed QP reduction technique than when using the same QP for the whole frame.

2. **Higher base QPs** - For the cases where the base QP is very high (e.g. QP = 37), the decisions at the RDO level in the contouring areas will most of the time choose the merge mode with no residual, in order to avoid spending too much bit rate resources on motion information. In these cases, the content in contouring areas in Inter frames is mainly repeated frame after frame until it is refreshed by an Intra frame. Since Intra frames are corrected by lowering the QP in vulnerable areas, the remaining frames will also not suffer from contouring. It is important to note that the perceptual quality in this case is not excellent due to other types

of artefacts, such as slight discontinuities in the boundaries of the contouring map, but it is much higher than without the reducing the QP in contouring areas.

The selection of the right QP to use in contouring areas is challenging, as this decision highly depends on the type of content being encoded. In the results further presented in this section, the contouring QPs for each video sequence tested were selected offline, after careful observation of the compressed sequences with different QPs. For some sequences, very low QPs were required to avoid the appearance of contouring artefacts in the decoded video sequences. For lower target qualities, where the difference between the base QP and the QP required for contouring artefacts prevention is very high, the relative bit rate increase is very high. Nevertheless, the visual quality in these cases is much higher since contouring artefacts become perceptually very annoying when no contouring prevention method is used.

When reducing the QP in contouring areas, it is important to recall that the λ remains the one corresponding to the base QP. For this reason, the RDOQ tool [77] available in the HEVC reference encoder will use the same λ to perform quantisation decisions at the coefficient level. In the techniques presented in this chapter, the RDOQ tool in HEVC was disabled in order to prevent all quantised levels being set to 0. Another HEVC tool that depends on λ is the SAO [52] in-loop filter. In order to prevent unexpected behaviours, SAO was also disabled in the results presented later in this section. It is important to note that both these tools could be adapted to work with the proposed contouring prevention solutions to possibly achieve further improvements.

4.2.2 Modified rate-distortion costs in contouring areas

A possible solution to overcome the slight visibility of contouring artefacts observed for lower base QP's is to foster the copying of previously encoded blocks as predictions in Inter frames to avoid completely flat blocks in the output video signal. This extension to the QP reduction method described in the previous subsection intends to force the RDO

process to select predictions from Intra frames that were already corrected by using a lower QP in the contouring areas. Figure 4.4 shows a visual example of how the proposed extension is applied.

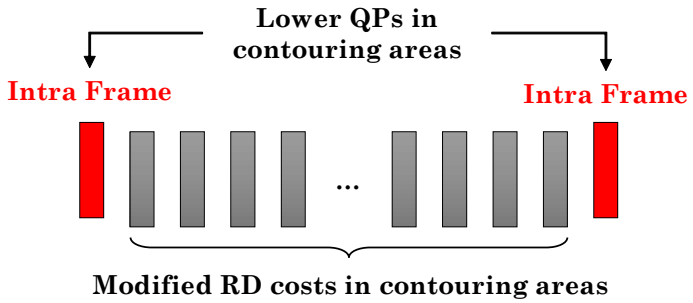


Figure 4.4: Frames where the proposed modified rate-distortion costs in contouring areas is applied.

In order to increase the direct copy of previously encoded blocks in previously encoded frames, a modification of the cost associated with the merge mode candidates is proposed. This modification increases the usage of the merge mode in contouring areas, especially when no residual signal is considered.

In contouring areas, the proposed modified cost computation for each merge mode candidate is given by

$$J = p \cdot D + \lambda \cdot R, \tag{4.1}$$

where D denotes the distortion introduced by the merge candidate being evaluated in terms of MSE, λ denotes the Lagrangean multiplier, R denotes the bit rate associated with the merge candidate being tested and p is a parameter introduced to intentionally reduce the weight of the MSE-based distortion. After extensive experiments, the parameter p was set to

$$p = \begin{cases} 0.1, & \text{CBF} = 0 \\ 0.7, & \text{otherwise} \end{cases}, \tag{4.2}$$

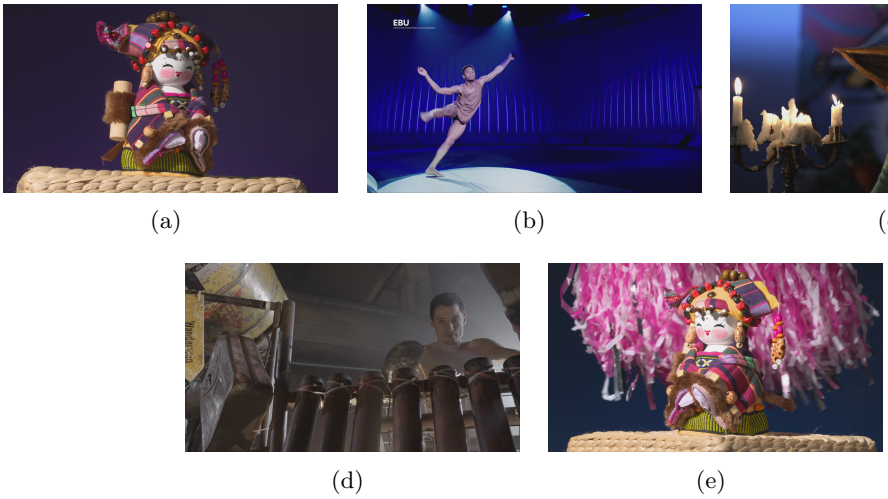
where CBF represents the Coding Block Flag for the merge candidate being tested. A CBF = 0 indicates that the residual after quantisation for that particular merge mode candidate being tested is null, meaning that the final reconstructed block will be directly given by the selected prediction samples, in what is referred to as "skip" mode (see Chapter 2 Section 2.3.3).

The parameter p was introduced based on the assumption that the computed MSE-based distortion is not well correlated with the visual quality observed in contouring areas and that copying areas of previously encoded frames results in a better subjective quality for these cases. By introducing p, the weight of the distortion in the cost computation is reduced, since MSE is not a reliable metric to assess the perceptual quality of the decoded video in contouring areas. This encourages the encoder to directly copy parts of previously encoded frames which were specifically encoded to preserve low amplitude details in contouring areas, either by direct copying previous frames (Inter case) or by performing quantisation with a lower QP (Intra case).

4.3 Performance evaluation

The performance of the techniques proposed in this chapter was evaluated using the first 100 frames of 5 different UHD sequences with bit depth of 8 bits per sample, 4:2:0 chroma format, 3840×2160 spatial resolution and frame rate of 50 and 60 fps. The test set is composed of content representative of broadcasting and all sequences are affected by contouring artefacts after compression. A sample image of each test sequence considered in this section is shown in Figure 4.5, to give an idea of the type of sequences where contouring artefacts might occur.

The base QPs used to define the rate-distortion points tested were 22, 27, 32, and 37, according to the common test conditions defined in [78]. The QPs used in the contouring areas were determined after careful evaluation of the visibility of contouring artefacts for the sequences tested. These QPs are shown in the first column of Table 4-A. All tests



(a) (b) (c)

(d) (e)

Figure 4.5: First frame of each original UHD test sequence. a) Ningyo; b) YoungDancers; c) CandleSmoke; d) ShowDrummer; e) NingyoPompoms.

were performed under the random access encoding configuration, typically associated with broadcasting scenarios.

In order to evaluate the bit rate increase associated with the described contouring artefacts prevention solutions, the QP reduction solution analysed in Subsection 4.2.1 (QPR in Table 4-A) and the proposed modified RD costs described in Subsection 4.2.2 (QPMC in Table 4-A) were implemented on top of the HEVC reference software HM 14.0. Table 4-A shows the bit rate increase associated with each solution with respect to the bit rate associated with the reference software. Figure 4.6 also shows the RD curves associated with HM 14.0 and both QP reduction techniques analysed for 4 of the selected test sequences.

In general, as shown in Table 4-A, very low contouring QPs are required to remove/reduce the visibility of contouring artefacts. This is especially problematic in the sequence YoungDancers, as there are significant bit rate increases when reducing the QPs in areas prone to contouring artefacts, especially when higher base QPs are used. It is important to emphasise though that the target qualities for higher QPs (e.g. 32 and 37) are associated with low bit rates and the visual quality in these cases is already signifi-

Table 4-A: Bit rate and PSNR comparison between the proposed solutions and the HEVC reference software

		QPR		QPMC	
Sequence	Base QP	Rate diff.	PSNR diff.	Rate diff.	PSNR diff.
CandleSmoke 50 Hz Cont. QP = 21	**22**	**1%**	**0.00**	**0%**	**-0.01**
	27	**3%**	**0.00**	**3%**	**-0.01**
	32	9%	0.00	9%	-0.01
	37	18%	0.00	18%	-0.01
NingyoPompoms 50 Hz Cont. QP = 25	**22**	**0%**	**0.00**	**0%**	**0.00**
	27	**0%**	**0.00**	**0%**	**-0.07**
	32	0%	0.00	0%	-0.07
	37	1%	0.00	1%	-0.06
Ningyo 50 Hz Cont. QP = 20	**22**	**2%**	**-0.01**	**-1%**	**-0.28**
	27	**6%**	**-0.03**	**4%**	**-0.16**
	32	12%	-0.03	11%	-0.09
	37	21%	-0.02	21%	-0.04
ShowDrummer 60 Hz Cont. QP = 22	**22**	**0%**	**0.00**	**-1%**	**-0.05**
	27	**3%**	**0.00**	**2%**	**-0.03**
	32	8%	0.00	8%	-0.04
	37	14%	-0.01	14%	-0.03
YoungDancers 50 Hz Cont. QP = 19	**22**	**2%**	**0.00**	**-1%**	**-0.09**
	27	**7%**	**0.00**	**5%**	**-0.04**
	32	25%	0.01	24%	-0.05
	37	51%	0.02	51%	-0.03

cantly damaged by other compression artefacts, such as blurring or blocking. Contouring removal methods might not be appropriate to be used in these cases since these kind of artefacts may be considered acceptable in these low bit rate scenarios, given the severity of other visually annoying compression artefacts. For this reason, the values corresponding to test points with lower QPs (higher qualities) are highlighted in Table 4-A, since the contouring prevention techniques described in this work are more relevant for these cases. The additional bit rate needed to prevent contouring artefacts is also dependent on the size of the areas identified by the contouring detection algorithm. For sequences like NingyoPompoms, where these areas are very small, most of the CTUs are encoded like in the HEVC reference software.

As seen in the RD plots in Figure 4.6, both techniques are able to keep the RD performance slightly lower than HM 14.0, meaning that the impact of removing contouring

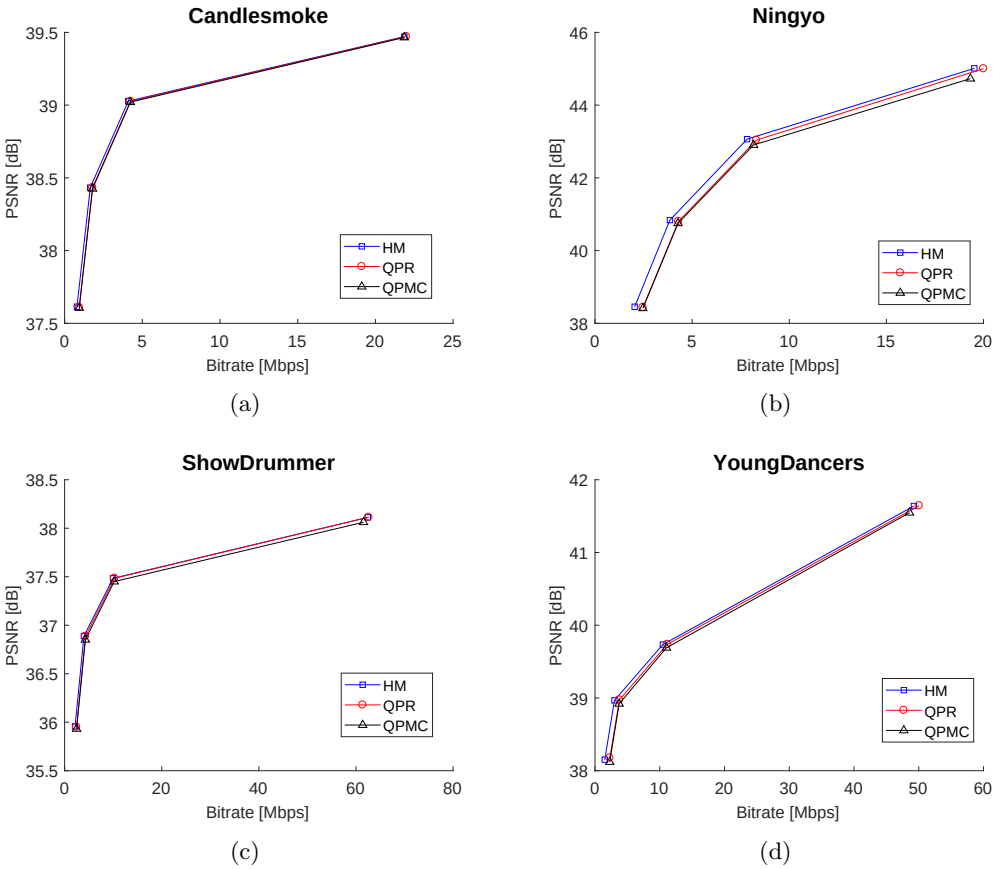


Figure 4.6: Rate-distortion performance analysis of the described methods for the sequences a) CandleSmoke, b) Ningyo, c) ShowDrummer and d) YoungDancers.

artefacts with these methods is not very high in terms of RD losses. Moreover, considering that the contouring techniques described in this work are mainly targeted for higher decoded video qualities (lower QPs), the RD loss for these cases is very low, as shown by the upper part of the RD curves in Figure 4.6. For the Ningyo sequence, the QPMC technique shows a worse RD performance than average since contouring areas are larger and the RD cost is modified in more CTUs to completely remove contouring artefacts, as previously mentioned.

Since PSNR does not fully reflect the visibility of contouring artefacts, a careful inspection of the decoded sequences is needed. The QPR technique is able to reduce or remove contours in most cases. For the Ningyo sequence, which is the most affected

by contouring artefacts, some slightly visible contours still appear in the decoded video for this technique, even though these contouring artefacts are significantly reduced when compared to the HM reference software. When the QPMC technique is used, the contouring artefacts are completely removed for this sequence, since most of the contouring areas in Inter frames are copied from previously encoded frames, which have been encoded to avoid contouring. Occasionally, for other sequences, other visual artefacts may appear, especially in the borders of the contouring maps. This is the case in the YoungDancers sequence where the accuracy of the contouring maps is not always perfect. The inaccuracy of the contouring maps in some sequences also affects the bit rate increase in all quality points. If the contouring detection algorithm fails to identify stationary background areas and erroneously identifies some textured areas, the effects of reducing the QPs in these areas will have much higher impact in the final bit rate. To better illustrate the benefits of the proposed contouring prevention techniques, Figure 4.7 shows an example of the same area of the Ningyo sequence shown in Figure 4.1 corrected with the QPMC method.

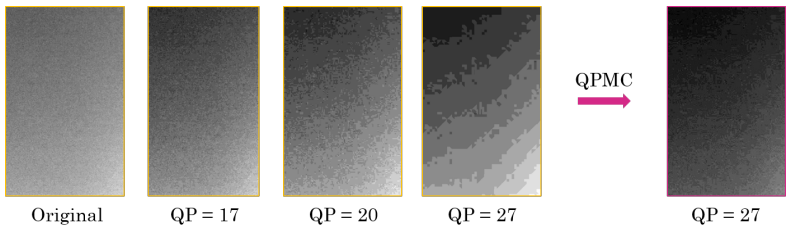


Figure 4.7: Example of an area of the Ningyo sequence encoded using different QPs and the result of the same area encoded with the QPMC method for QP 27.

Finally, in order to compare the proposed solution with another contouring prevention method from the literature, Table 4-B reports the BD-rate values for both the QPMC and the Adaptive Dead Zone Adjustment (ADZA) method [85], with respect to the HEVC reference software. The performance of these contouring prevention methods is also assessed using the Pixel Variation Preservation (PVP) score [85], specifically proposed to assess the visibility of contouring artefacts. PVP quantifies the preservation of pixel variations over contour prone blocks. The values of PVP range from 0 to $+\infty$, where the

higher the score the less visible contouring artefacts are. The results of the PVP score are depicted in Figure 4.8.

Table 4-B: BD-rates of QPMC and ADZA in comparison to the HEVC reference software

Sequence	QPMC	ADZA
Candlesmoke	12.4%	2.3%
NyngyoPompoms	9.6%	5.0%
Ningyo	13.9%	7.9%
ShowDrummer	15.7%	2.0%
YoungDancers	12.7%	5.2%
Average	12.9%	4.5%

From the results in Table 4-B, it can be observed that the impact in terms of BD-rate penalty associated with ADZA is smaller than QPMC. This can be explained by the fact that in QPMC, a very low QP needs to be selected in contouring areas to completely remove contouring artefacts, even for lower target qualities (e.g. base QP = 37). Again, this reinforces that the QPMC technique is more suitable for high target qualities, where the difference between the base QP and the contouring QP is not significantly high.

As for the results of the PVP score in Figure 4.8, both techniques achieve higher PVP scores than the anchor, which means that both succeed in preserving the small details in contouring areas that are discarded after compression. Due to the QP reduction used in contouring areas, the QPMC method is able to preserve these details consistently better than the ADZA technique.

4.4 Conclusion

To sum up, this chapter presented and analysed two techniques proposed to prevent the appearance of contouring artefacts in compressed UHD sequences by reducing the quantisation parameter in areas prone to contouring artefacts. The first technique consisted of reducing the QP in contouring-prone areas for all types of frame in a video sequence. The details and challenges associated with the implementation of this technique were

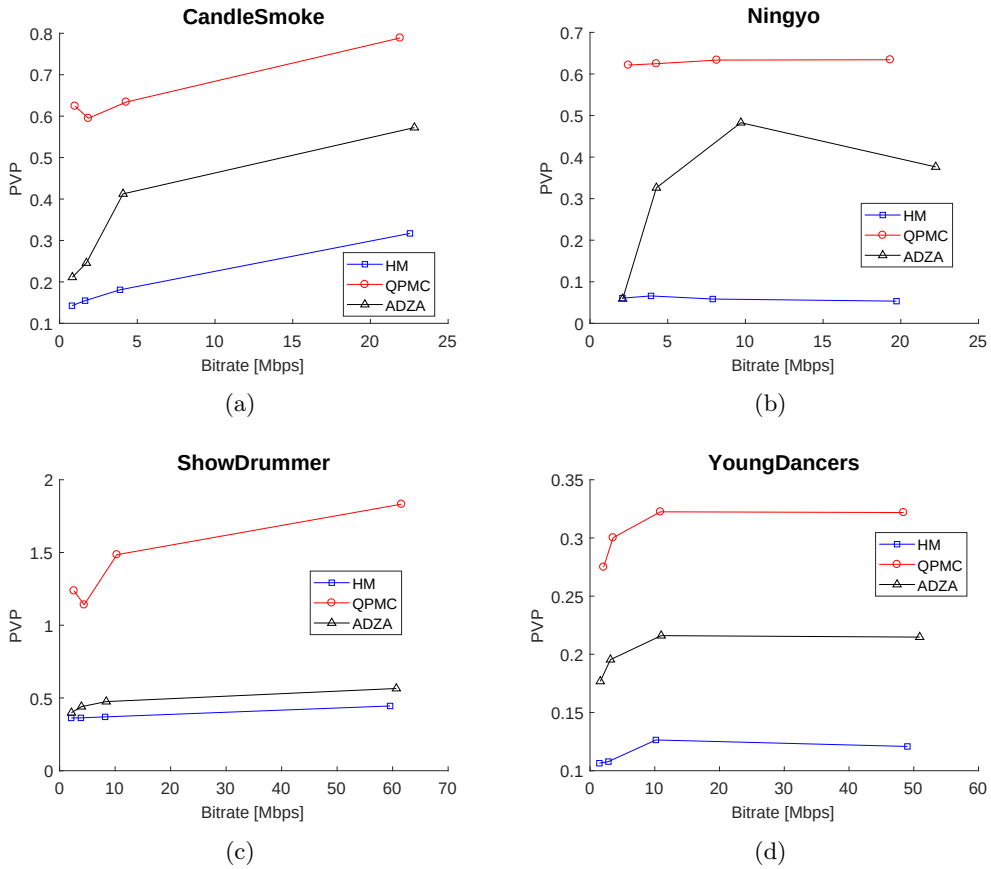


Figure 4.8: PVP scores for QPMC and ADZA. a) CandleSmoke; b) Ningyo; c) Showdrummer; d) Youngdancers

described, which led to the design of the second contouring prevention method. This method aimed to further improve the reduction of contouring artefacts by modifying the RD costs associated with the merge mode candidates in Inter frames.

The perceptual quality of the observed results was considered to be satisfactory since most contouring artefacts were removed from the decoded sequences or made less visible. Both solutions analysed mainly target higher bit rate scenarios, since for lower bit rates contouring artefacts may be considered less significant when compared to other visual degradations. Finally, a comparison with another contouring prevention method in the literature showed that the QP reduction method with modified costs is able to preserve better the pixel variations in contouring areas than the method from the literature,

according to the PVP score. However, higher bit rate increases are introduced for lower qualities, which reinforces the conclusion that the proposed method is more suitable for higher target qualities, where the associated bit rate increase is lower.

Possible improvements can be added to the described techniques, notably by refining the contouring areas detection process to improve the accuracy of the contouring maps. Additionally, a method to automatically select the adequate contouring QPs according to the video content being encoded is also needed to integrate the proposed techniques into practical video encoding applications.

Chapter 5

Improved Intra coding techniques beyond HEVC

The techniques described in previous chapters aim to improve the coding performance of HEVC encoders through perceptual optimisations. All encoding schemes presented in Chapters 3 and 4 generate a bit stream perfectly compliant with the HEVC standard. Differently, in this chapter, the proposed coding tools aim to go beyond what is defined in the HEVC specification, targeting to improve, in particular, its Intra prediction performance. Therefore, the techniques proposed in this chapter require changes to the syntax of HEVC, producing this way bitstreams that are not compliant with the standard. It is important to note that HEVC is the state-of-the-art video coding standard, which already provides a large set of advanced video coding tools that altogether can achieve a remarkable video compression performance [29]. For this reason, designing new algorithms to further improve the coding efficiency of HEVC is a challenging task.

The Intra coding process is an essential part of any video compression system. As Intra-coded frames do not require any information from previously encoded content, they are essential to prevent error propagation and provide random access to a video stream. Furthermore, improving the efficiency of Intra frames can bring higher performance also

to Inter prediction, due to better reference frames used for motion estimation.

In this chapter, two approaches to improve HEVC's Intra coding performance are proposed. The first one is based on using artificial spatial patterns to reduce the Intra prediction error and consequently reduce the amount of residual information that needs to be conveyed in the compressed bit stream. The second one is based on the so-called combined Intra prediction technique, which uses information from within the block being encoded to improve the accuracy of Intra predictions.

The remainder of this chapter is organised as follows. Section 5.1 briefly describes some relevant state-of-the-art Intra coding methods. Section 5.2 describes the details of the proposed Intra prediction improvement technique based on artificial spatial patterns. Section 5.3 describes in detail the so-called Combined Intra Prediction (CIP) scheme [86] and two proposed Intra prediction methods based on this approach to improve the Intra coding performance of HEVC. Finally, Section 5.4 presents the performance evaluation of all Intra coding tools proposed in this chapter.

5.1 Background work

Several tools were proposed to improve the performance of Intra prediction in video compression solutions. Bi-directional Intra prediction [87], for example, was proposed to improve the accuracy of the Intra prediction framework adopted in the AVC standard. This technique consists of combining the two prediction blocks obtained from two different prediction directions using a weighted average. The weights are given by weighting matrices that are defined based on the Intra prediction modes being combined and on the position of the samples inside the block. More recently, another technique to combine Intra prediction direction modes was proposed for AVC and HEVC based on the computation of adaptive weights at the encoder that can be recovered at the decoder [88].

Other techniques relying on template matching [89] were also proposed in the literature as a way to improve Intra prediction. These techniques comprise searching in the previously encoded parts of the picture for blocks with a surrounding top-left region, called a template, similar to that of the block to predict. The block with the best matching template is then used for prediction. This search can be replicated at the decoder to find exactly the same best predictor and therefore no additional signalling is needed. Other variants of template matching, such as priority-based template matching [90], neighbour embedded methods [91] or image inpainting-based techniques [92] have also been proposed to better explore the information on the causal neighbourhood of the block. More recently, the Intra block copy mode [93] adopted in the HEVC extensions for Screen Content Coding [27] was also modified based on template matching to make this technique also relevant for camera content [94]. The drawback of all these solutions is the fact that the template search needs to be carried out at the decoder as well, increasing significantly the complexity at the decoder side.

More relevant to the work presented further in Section 5.2, the technique proposed in [95] relies on artificial information to improve Intra prediction performance. This technique follows a frequency domain prediction workflow, previously proposed in the literature [96], where both the original and the prediction blocks are transformed separately prior to the computation of the residual. In [95], the suppression of some specific residual coefficients is performed, followed by the addition of predefined artificial coefficient values. These two processes are used to compensate for inaccuracies of Intra prediction in the frequency domain. As explained further in Section 5.2, the frequency domain prediction process has a relevant impact in the complexity of the decoder.

Taking into account that low decoder complexity is important in typical video compression applications, the Multi-Parameter Intra (MPI) prediction tool [97] was proposed to be applied on top of the Intra prediction process specified in HEVC. This method consists of averaging the prediction samples generated by the Intra prediction mechanism in HEVC with the immediately left and/or top predicted samples using appropriate weights.

The usage of MPI is signalled at the CU level, using up to 2 bits to define which samples (only left, only top or top and left) should be used in the average computation. As described later in Section 5.3, MPI is used in combination with the improved CIP technique proposed in this chapter to further improve Intra prediction and consequently the Intra coding performance of HEVC.

5.2 Intra coding using artificial patterns

This section presents a technique that aims to improve the accuracy of Intra predictions by using a special set of spatial patterns derived from preliminary statistical observations of some key Intra coding elements. These preliminary observations were also useful to better understand Intra prediction in HEVC and to understand in which areas possible improvements may have a higher impact in improving Intra coding performance.

The proposed approach follows the rationale used in [95], where artificial patterns were used to compensate for the inaccuracies of Intra prediction in the frequency domain. In this context, the term "artificial patterns" refers to predefined image patterns which are content independent. As mentioned in the previous section, this approach is conceptually different from the traditional approach typically followed in modern video compression standards. In a typical video encoder, residual blocks are computed in the spatial domain by computing the difference between the original block and the prediction block. The obtained residual is then transformed, quantised and entropy encoded before being transmitted to the decoder in the bit stream. The inverse operations are performed at the decoder to obtain the reconstructed version of the block. Conversely, in [95] and other frequency domain prediction techniques, the computation of the residual signal is done in the frequency domain. In this case, both the original and the prediction blocks are first transformed separately. The obtained transformed residual block is then quantised and sent to the decoder. At the decoder side, quantised residuals are dequantised and the prediction block is transformed. The transformed prediction and the dequan-

tised residuals are then added to generate the transformed reconstructed block, which is then inverse transformed to obtain the final reconstructed block.

It is important to note that using the described frequency domain residual computation framework requires an extra transform operation both at the encoder and at the decoder side, with respect to the conventional hybrid workflow. This is because the prediction signal needs to be transformed in the former case. This naturally introduces significant complexity increase at the decoder side, which is not desirable. However, the Intra coding performance gains reported in [95] are significant.

In [95], several manipulations in the transform coefficients of the prediction take place to minimise the error between the predicted coefficients and the original ones. Briefly, these manipulations in the frequency domain have two distinct stages:

1. First, some selected high frequency coefficients of the prediction signal are suppressed. The selection of the coefficients to be suppressed is performed according to the size of the Intra prediction block and the selected Intra prediction mode.

2. The suppressed coefficients are then replaced by synthetically generated values from adequately generated look-up tables. Several coefficient values are tested and the best configuration in terms of RD cost is selected. The information relative to the indexes of the look-up tables has to be signalled in the bit stream, so that the same Intra predictions can be replicated at the decoder using the same look-up tables used at the encoder.

The proposed tool in this section aims to perform a similar processing to the prediction signal in the spatial domain, in order to retain the compression gains achieved in [95] without the extra complexity needed, which can be especially problematic for the decoder.

5.2.1 Preliminary observations

As a first stage towards improving the performance of Intra prediction methods in the spatial domain, some statistics that quantify the accuracy of the Intra predictions generated by the HEVC reference software HM 16.6 were collected, in order to better understand the characteristics of the Intra prediction residuals. For this, a set of 9 UHD video sequences and 1 HD sequence were encoded using the HEVC reference software HM 16.6 in the all Intra configuration. The differences between the luma samples of the original images and the luma Intra predictions generated and selected by the encoder as the best Intra prediction option were computed. Figure 5.1 shows the average Intra prediction error per sample position observed in 32×32 blocks.

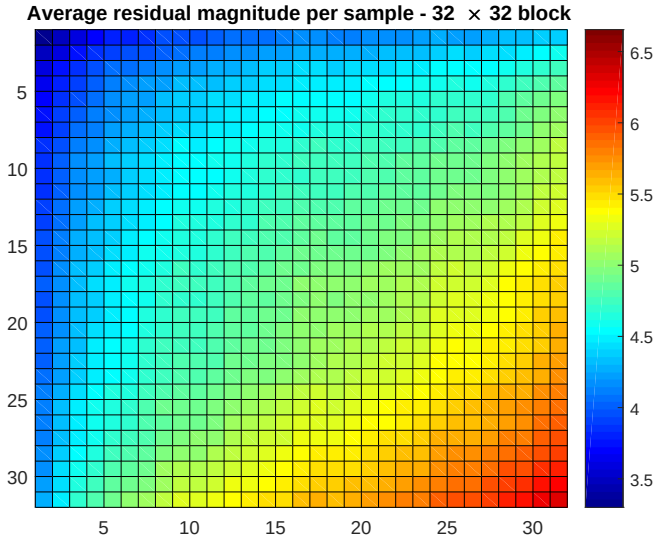
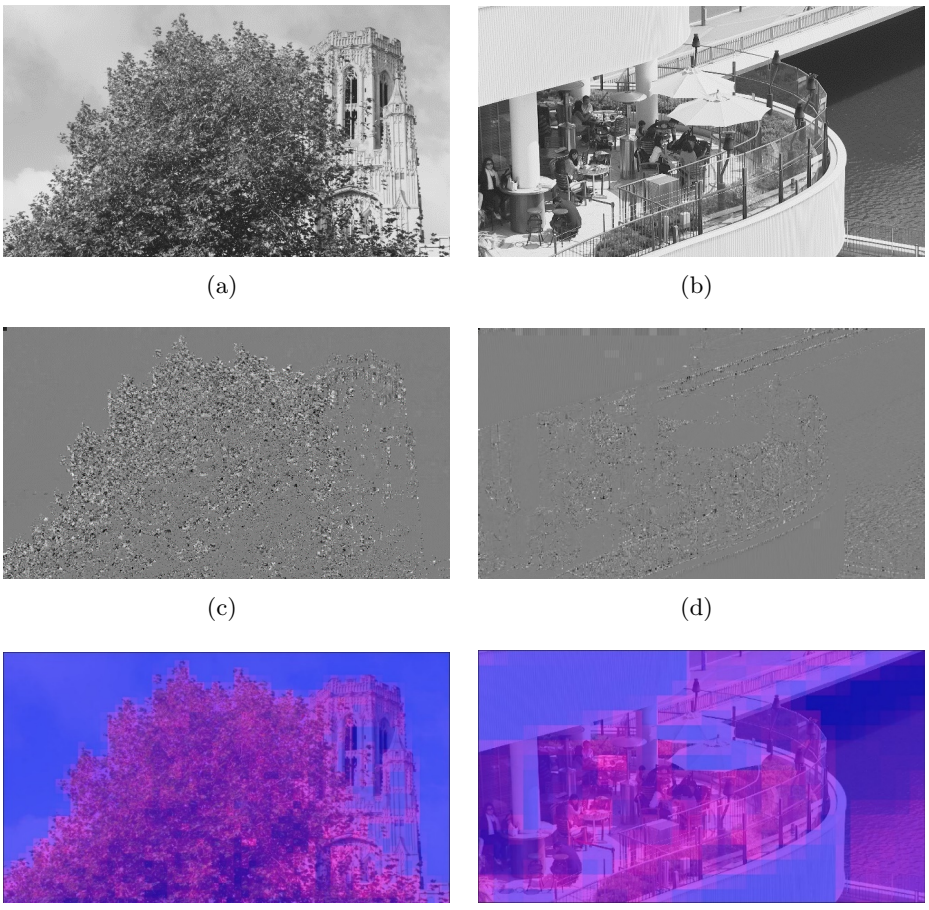


Figure 5.1: Average prediction error per sample position for 32×32 prediction blocks.

As seen in Figure 5.1, higher prediction errors are observed for prediction samples located close to the bottom right corner of the block. This is expected since the spatial correlation between these samples and the reference samples is lower due to a higher distance from the reference samples. This behaviour is consistent for the remaining prediction block sizes and suggests that possible improvements in the Intra prediction

process should be focused on the prediction of the areas of the prediction block that are further away from the reference samples used to build the prediction.

Additionally, Figure 5.2 shows a visual example of the prediction error and distribution of bits spent to encode the first frame of two different video sequences.



(a) (b) (c) (d) (e) (f)

Figure 5.2: a, b) Original images; c, d) Prediction error; e, f) Distribution of the bits spent after encoding each image, where red areas correspond to areas where more bits are used.

As seen in Figure 5.2, Intra prediction in HEVC is very accurate when the content being encoded is flat or when the edges in the surrounding blocks are propagated to the current one. When the content being encoded is more textured and less regular, less accurate predictions are generated, producing higher residuals and therefore increasing the bit rate needed to represent the video content.

5.2.2 Intra prediction improvements based on artificial patterns

Given the statistics and observations shown in the previous subsection, a method for improving Intra prediction in HEVC was designed. The main goal of this method was to explore the possibility of building a set of spatial patterns that can be added to the Intra prediction blocks in the spatial domain in order to improve the prediction accuracy in textured regions. The starting point of this work was to collect the same prediction error statistics shown in Figure 5.1 for all block sizes and all prediction modes available, in order to understand what kind of information is missing in the Intra predictions generated in HEVC. It is important to recall that this prediction error corresponds to the difference between the original image and the Intra predictions selected by the encoder for Intra coding. Therefore, these differences, from now on referred to as absolute residual, indicate how far the predictions generated by HEVC are from being optimal. Figure 5.3 shows colour maps of the average absolute residuals obtained according to the position of samples in the block for the most used HEVC Intra prediction modes in 32×32 Intra-coded blocks.

The patterns in Figure 5.3 can be interpreted as the average of the information that is missing to make HEVC's Intra prediction perfect. Therefore, these patterns may be considered as possible additional artificial information that can be combined with the predictions generated by the normal HEVC Intra prediction mechanism to improve the Intra compression performance of coding schemes beyond HEVC.

The average patterns similar to the ones shown in Figure 5.3 for each block size and prediction mode were classified into 3 different categories, in order to understand if the predictions were, in general, underestimating or overestimating the original luma values. The following categories were defined for this purpose:

- **Positive** - The absolute residual for a given block is considered to be mostly positive if more than 75% of its absolute residual values are positive.

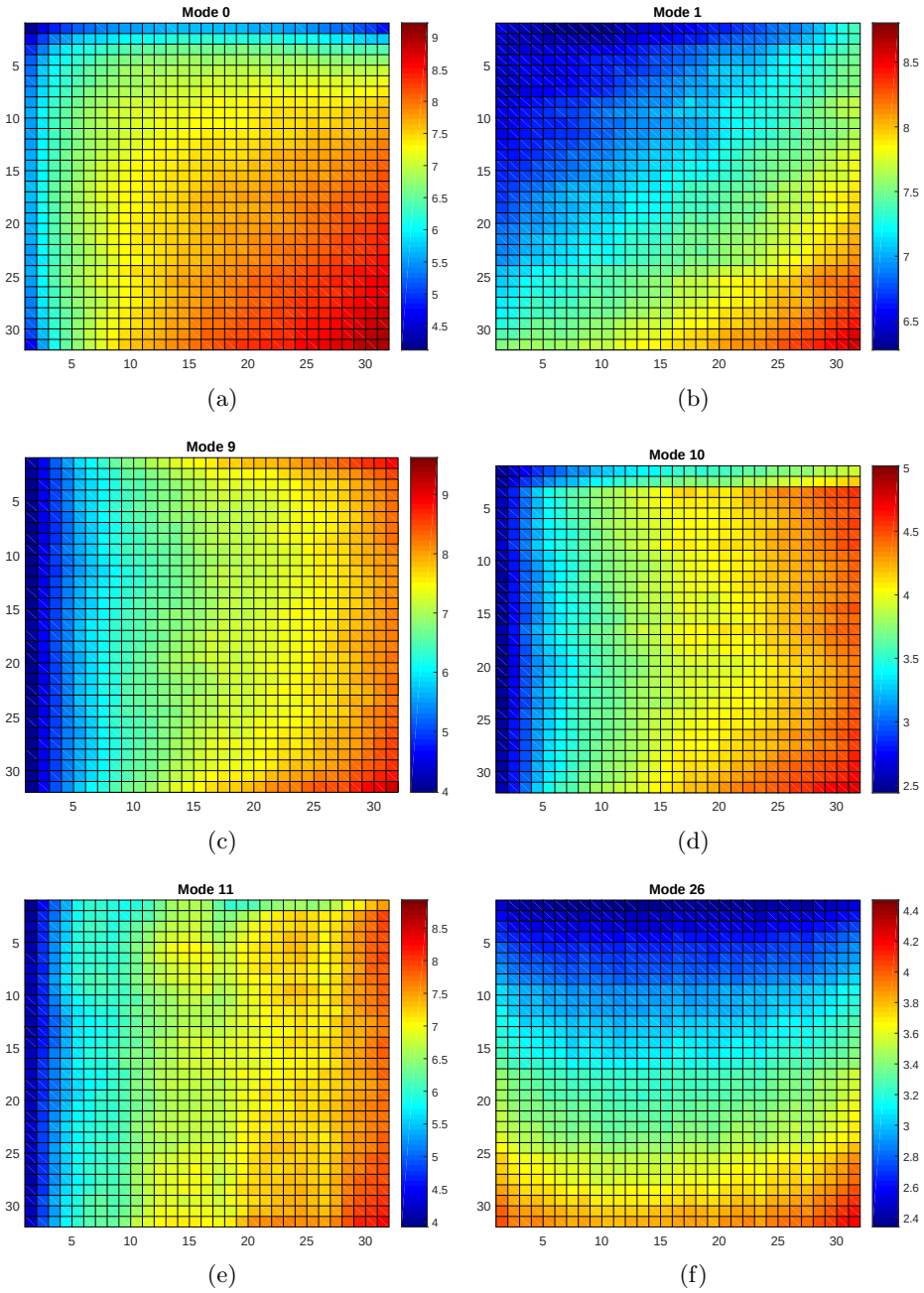


Figure 5.3: Average absolute residual patterns for the most used Intra prediction modes.

- **Negative** - The absolute residual for a given block is considered to be mostly negative if more than 75% of its absolute residual values are negative.

- **Mixed** All other blocks that do not show a significant majority of positive or

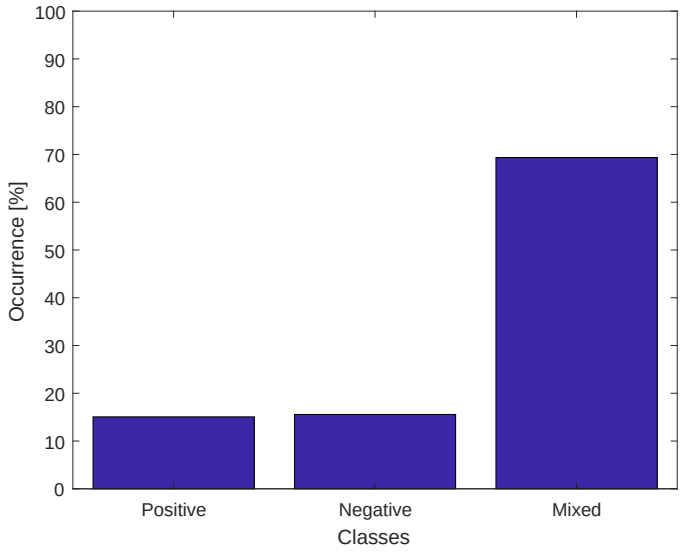


Figure 5.4: Percentage of blocks belonging to each class of absolute residuals.

negative absolute residual values are classified as mixed.

The percentage threshold of 75% that defines the classification was considered suitable for a preliminary study of the proposed technique. The plot in Figure 5.4 shows the average distribution of each of the aforementioned classes for the set of sequences used to generate the patterns in Figure 5.3. The percentage of occurrences in Figure 5.4 corresponds to the average of all block sizes and all Intra prediction modes.

Figure 5.4 reveals that around 30% of the Intra prediction blocks show a clear majority of either positive or negative absolute residuals. The blocks that fall into the class Positive are blocks where HEVCs Intra prediction in most cases underpredicts the original blocks. This means that in these blocks, additional information needs to be added to the prediction to improve it. Conversely, for blocks included in class Negative, the prediction values are mostly higher than the original block and therefore some information needs to be subtracted from the prediction.

Given the results of the aforementioned classification, an improved Intra prediction mechanism was designed where the encoder tests the possibility to add or subtract a

given pattern to the normal prediction blocks generated by the normal HEVC Intra prediction. The patterns used to improve the predictions were defined based on the reported observations. Different patterns were used for different block sizes and for different Intra prediction modes. For a given block size and Intra prediction mode, the pattern for adding (positive pattern) was computed as the average of the absolute residuals for the blocks classified as belonging to class Positive for that particular block size and mode. Similarly, the pattern for subtracting (negative pattern) was computed based on the blocks falling into the Negative class.

It is important to note that in HEVC, an Intra predicted block is generated for each TU according to the Intra prediction mode signalled for the corresponding PU. Therefore, to enable the additional possibility of adding or subtracting a pattern to the prediction, the encoder simply needs to additionally signal at the PU level if a pattern was added to the prediction, subtracted from the prediction, or if no pattern was used to improve the prediction (normal HEVC Intra prediction). A new parameter was therefore created and added to the bit stream to signal this information. This parameter is entropy encoded with CABAC after applying a truncated unary code binarisation as shown in Table 5-A.

Table 5-A: Binarisation of the parameter used to signal the usage of a spatial pattern.

Operation	Binary code
No pattern	0
Add pattern	10
Subtract pattern	11

After some preliminary tests, it was observed that there was a significantly higher usage of the proposed method for bigger blocks (16×16 and 32×32). Therefore, the proposed mechanism was restricted to be used only for these block sizes, as for smaller blocks the improvements in prediction quality did not compensate for the associated additional signalling overhead. The performance results obtained with the proposed technique are shown in Section 5.4.1. In short, as can be further verified in Section 5.4.1, the performance of this method does not encourage further study for now, as the coding

gains achieved are marginal. Nevertheless, the study conducted on the characteristics of the Intra prediction residual was important as it provides a better understanding of the areas where possible improvements can be made. Taking this into account, a different approach was investigated as described in the next section.

5.3 Improved combined Intra prediction

In this section, another Intra prediction technique is proposed to improve the overall Intra coding performance in HEVC. Differently from the method proposed in Section 5.2, this method comprises an alternative additional technique for building Intra predictions and the respective reconstruction process, based on the so-called CIP approach. The following subsections describe the original CIP scheme [86] and two proposed Intra prediction methods based on CIP to improve Intra coding performance in HEVC.

5.3.1 Combined Intra prediction

The main objective of CIP is to use not only information extracted from the surrounding of the current block, but also information from within the current block. In particular, CIP consists of generating Intra prediction for a given block using a combination of two different sources of prediction: an Outside-Block Prediction (OBP), and an Inside-Block Prediction (IBP). The OBP is generated using reference samples extracted from the close surroundings of the current block, and can be generated using any of the existing Intra prediction techniques. On the other hand, the IBP is generated using samples from within the current block. Figure 5.5 illustrates an example of the samples involved in the construction of CIP.

As depicted in Figure 5.5, the CIP samples are computed by combining the predictions p_{OB} and p_{IB}, obtained using OBP and IBP, respectively. The outside-block component is given by the conventional HEVC Intra prediction that is entirely based

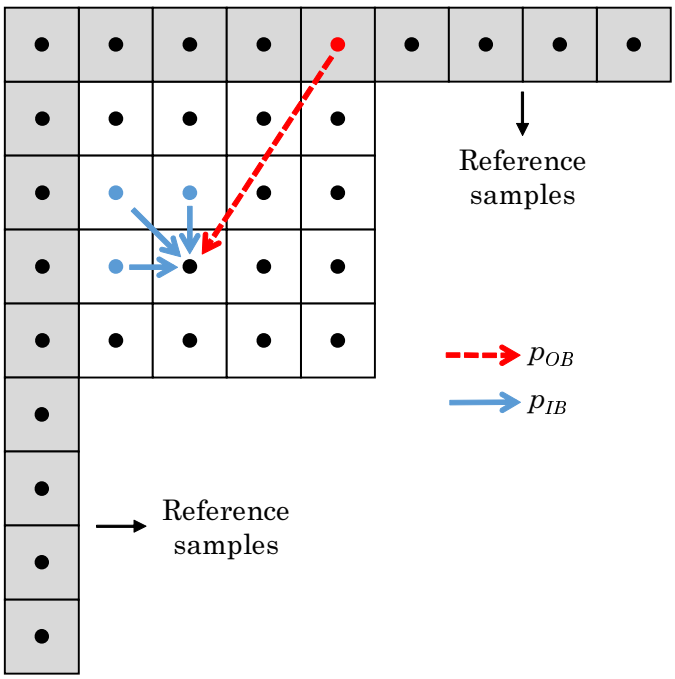


Figure 5.5: Samples involved in the computation of CIP.

on the neighbouring reconstructed samples. At the encoder, during Intra prediction, the inside-block component is computed using samples from inside the original block, according to:

$$p_{IB}(i,j) = \frac{a \cdot x(i-1,j) + b \cdot x(i,j-1) + c \cdot x(i-1,j-1)}{a+b+c}, \tag{5.1}$$

where $x(i,j)$ represents the original samples of the video sequence corresponding to the positions with indexes i and j in the block being predicted and the values a, b and c are used to control the weight each original sample has in the average computation. In other words, for a given sample in position (i,j) inside the current block being predicted, the IBP is given by a weighted average of the original samples located at its top, left and top-left positions. The same weights used in [86] are used for this purpose, i.e. $a = 3$, $b = 3$ and $c = 2$. Notice that when predicting samples in the top-most row (left-most column) of the block, the reference samples at the top (left) of the block are used instead of the original samples to compute the IBP.

The final CIP value is obtained by a weighted average between the OBP, given by the HEVC Intra prediction values, and the IBP, computed according to:

$$p_{CIP}\left(i,j\right) = \frac{\alpha \cdot p_{OB} + \beta \cdot p_{IB}}{\alpha + \beta}, \tag{5.2}$$

where $\alpha = 6$ and $\beta = 2$ define the weights given to each of the contributions to the final CIP value.

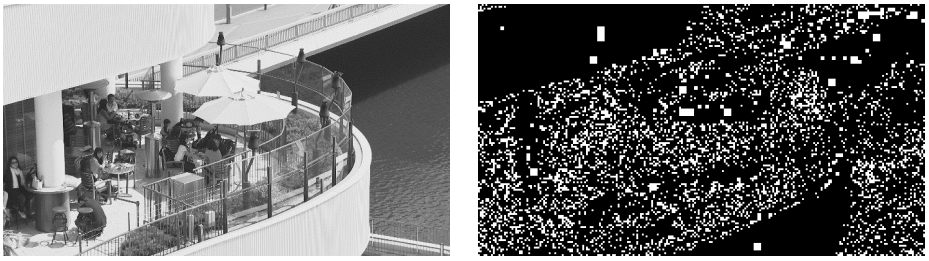
The computation of the IBP in Eq. (5.1) requires access to the original picture samples $x\left(i,j\right)$, which are only available at the encoder side. Such IBP based on the original samples is used to compute the residuals $r\left(i,j\right)$, which are then encoded and transmitted in the bit stream. At the decoder side, such residuals are uncompressed to obtain the decoded residuals $\tilde{x}\left(i,j\right)$. Then the sample at the top-left position in the current block is predicted as $\tilde{p}_{CIP}\left(0,0\right)$. As already illustrated, this process only involves using information from the reference samples, which means $\tilde{p}_{CIP}\left(0,0\right) = p_{CIP}\left(0,0\right)$. The reconstructed sample is then computed as $\tilde{x}\left(0,0\right) = \tilde{r}\left(0,0\right) + \tilde{p}_{CIP}\left(0,0\right)$. The value $\tilde{x}\left(0,0\right)$ is then used instead of $x\left(0,0\right)$ to compute $\tilde{p}_{CIP}\left(0,1\right)$ as in Eq. (5.1). The process is repeated for all samples $\tilde{p}_{CIP}\left(i,j\right)$. Hence, apart from the top-left sample, the CIP obtained at the decoder side is different from the one used at the encoder side. In order to avoid mismatches between encoder and decoder and maintain synchronisation, the IBP computation must be performed also at the encoder side using the reconstruction samples, in order to obtain $\tilde{p}_{CIP}\left(i,j\right)$.

This drift between the CIP used during the Intra prediction and the actual reconstruction process propagates only inside the block being encoded. It is important to note that this effect did not reveal any negative influence in the subjective quality of the decoded video sequences, after large scale testing with CIP. However, higher performance can be achieved if this drift effect is reduced. This is because the computation of the residuals which are then encoded and transmitted in the bit stream is based on an inaccurate version of the CIP, different from the one actually used during the reconstruction

process. In the following, an improved version of CIP is presented, aimed at reducing such drift, hence enhancing the performance of the technique.

5.3.2 Improved CIP using drift-free border prediction

The CIP technique is based on the assumption that the reconstructed samples at the decoder are similar to the original samples and therefore the small mismatch resulting from this difference is compensated by the enhanced prediction CIP can produce. The improvement brought by CIP is especially relevant for blocks containing samples that are less correlated with the neighbouring samples used as reference. This is illustrated in Figure 5.6 where a binary map shows the blocks where the CIP mode is selected by the encoder.



(a) (b)

Figure 5.6: a) Luma component of the first frame of the BQTerrace sequence; b) Map showing the blocks where CIP was selected (in white) or not (in black).

In Figure 5.6, the areas in white show the areas where CIP is used while the areas in black show the areas where HEVC Intra prediction is selected. It can be observed that CIP is selected in textured areas where the content inside the blocks is less correlated with their surroundings. These areas, as shown in Figure 5.2, are also the areas where more bits are spent for Intra coding.

The small mismatch between the prediction samples generated by the encoder and the decoder when CIP is used naturally reduces the benefits of using CIP. This is specially observed when encoding with higher QPs, where the resulting reconstructed samples

inside the block significantly differ from the original samples. In order to reduce the impact of this mismatch and reduce the drift in the predictions generated by the encoder and the decoder within the prediction blocks, a change in the way CIP is computed at the borders of the prediction blocks is proposed.

In the original implementation of CIP, at the borders of the blocks being predicted, the IBP is computed by replacing the original samples with the reference samples when these are available. When $i = 0$ and $j = 0$, the IBP can be computed entirely based on the reference samples. In this case, no mismatch is introduced as the same reference samples are available both at the encoder and decoder. However, the same does not happen for the remaining samples of the top row and left column of the block being predicted. In these cases, one of the samples used to compute the IBP is extracted from within the block and therefore the corresponding original sample has to be used at the encoder, generating a drift between encoder and decoder (see Figure 5.7).

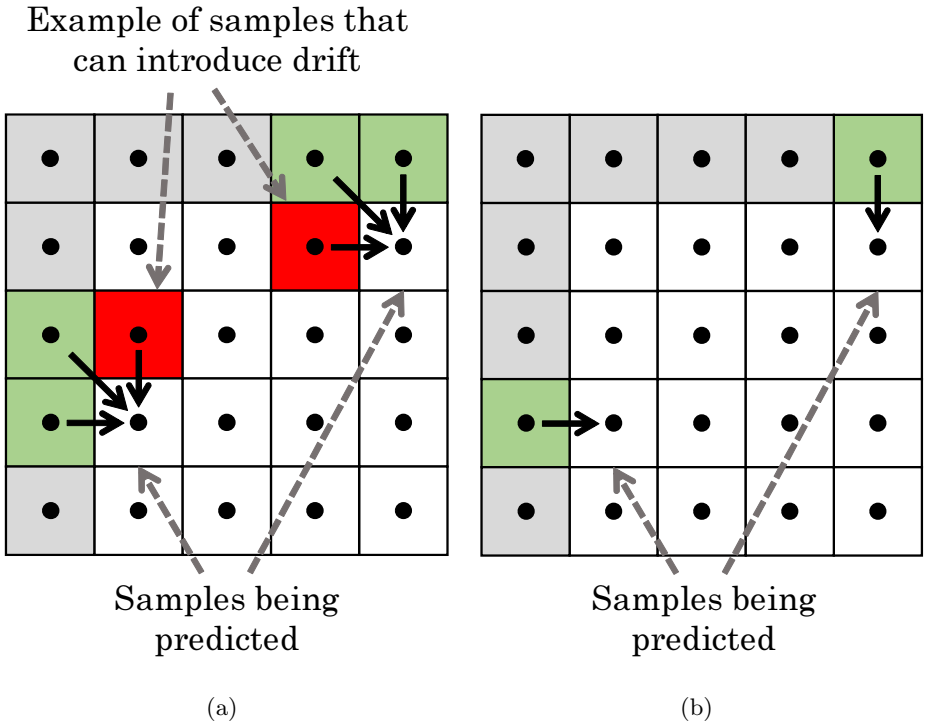


Figure 5.7: a) Samples involved in the computation of p_{IB} in the original CIP scheme; b) Proposed modification in the computation of p_{IB}.

A different approach is proposed to compute the IBP in the border of the block. For the samples in the top-most row and left-most column, the proposed IBP is given by

$$p_{IB}\left(i,j\right) = \begin{cases} \tilde{x}\left(i-1,j\right), & i=0, j>0 \\ \tilde{x}\left(i,j-1\right), & j=0, i>0 \end{cases} \tag{5.3}$$

where $\tilde{x}\left(-1,j\right)$, $\tilde{x}\left(i,-1\right)$ and $\tilde{x}\left(-1,-1\right)$ are the reconstructed samples from the neighbouring blocks used as reference samples in conventional Intra prediction. This approach only eliminates the drift in the borders of the blocks predicted with CIP, namely $\tilde{p}_{CIP}\left(i,j\right) = p_{CIP}\left(i,j\right)$, if $i=0$ or $j=0$. All other samples in the block still need to be predicted using the original samples during Intra prediction at the encoder and the reconstructed samples during the reconstruction process both at the encoder and decoder. Nevertheless, as the drift propagates along the block while the combined Intra reconstruction operation takes place, using the proposed technique results in a significant reduction in the drift throughout the whole block. The results shown in Subsection 5.4.2 confirm the positive impact this modification has on the overall rate-distortion performance of CIP.

5.3.3 Improved CPI + MPI

When using CIP, the propagation of the drift throughout the block has a higher negative impact in RD performance when considering larger blocks (such as 16×16 or 32×32 luma samples), since the drift propagates throughout a larger area. Conversely, in smaller blocks where the drift has a smaller effect, CIP provides higher benefits. The MPI approach, proposed in [97], consists of performing a similar prediction operation to CIP but exclusively based on prediction samples, thus avoiding the prediction/reconstruction drift. The method is instead less effective than CIP in smaller blocks, because neighbouring prediction samples are generally less helpful in improving the prediction than reconstructed samples.

Therefore, an alternative approach is also proposed to combine the two techniques,

referred to as Improved CIP + MPI. The combined solution consists of using the proposed Improved CIP when the Intra prediction is performed on block sizes of 4×4 and 8×8, since the negative effect of drift propagation within the block is less accentuated in smaller blocks. When Intra prediction is performed for 16×16 or 32×32 blocks, MPI is used instead.

The proposed combination of Improved CIP + MPI only requires 1 extra bit for signalling per CU, to indicate whether to use HEVC Intra prediction, or to use an alternative Intra prediction technique (Improved CIP or MPI). In case the latter is preferred, CIP is applied when performing Intra prediction on smaller blocks (4×4 or 8×8) and MPI is applied when performing Intra prediction on larger blocks (16×16 and 32×32). The encoder complexity associated with this combination of CIP and MPI is approximately the same as using Improved CIP or MPI alone, as only one these techniques is additionally tested by the encoder for each block size.

5.4 Performance evaluation

In this section, the performance of all Intra coding tools proposed in this chapter is evaluated. Subsection 5.4.1 reports the performance results for the Intra improvement technique based on spatial patterns, described in Section 5.2, while Subsection 5.4.2 reports the performance of the CIP-based tools proposed in Section 5.3.

5.4.1 Intra coding using artificial patterns

The improved Intra prediction mechanism proposed in Section 5.2 was implemented on top of the HEVC reference software HM 16.6 [98]. The average patterns used for each mode and block size were generated based on the statistics reported in Subsection 5.2.1, after encoding a set of 9 UHD video sequences and 1 HD sequence in all Intra configuration with QPs 22, 27, 32, 37 [78] using HM 16.6. These patterns are the same

for every sequence tested and are stored in memory both at the encoder and decoder.

The performance of the proposed Intra prediction scheme is summarised in Table 5-B using the BD-rate as a measure of compression performance improvement for the all Intra coding configuration defined in [78]. Additionally, Table 5-C shows the percentage of blocks for all encoded sequences where the encoder selected the proposed technique by using a spatial pattern to improve Intra prediction (either adding or subtracting) and Table 5-D shows the percentage of execution time associated with the proposed method, with respect to the reference HM 16.6.

Table 5-B: Performance of the proposed Intra prediction enhancement method based on spatial patterns.

	Y	U	V
CentrallineCrossing	0.0%	0.0%	0.0%
HomelessSleeping	0.0%	-0.5%	0.1%
LampLeaves	-0.3%	0.0%	0.0%
Manege	-0.1%	0.1%	0.1%
NingyoPompoms	-0.2%	-0.1%	0.0%
Petitbato	0.0%	0.1%	0.0%
Sedof	0.0%	0.0%	0.0%
TreeWills	-0.3%	0.1%	0.1%
BQTerrace	0.0%	0.0%	0.0%
Fountains	-0.1%	0.1%	-0.1%
Average	-0.1%	0.0%	0.0%

Table 5-C: Usage of the proposed spatial pattern-based Intra prediction improvement.

Size	16×16	32×32
Usage:	10.16%	8.82%

Table 5-D: Complexity associated with the proposed spatial pattern-based Intra prediction improvement with respect to HM 16.6.

Encoding	Decoding
107%	100%

From the results in Table 5-B, it can be concluded that only very small BD-rate gains can be achieved with the proposed method. From the several tests performed to evaluate this approach, it was concluded that most improvements come from the fact that this

technique is only applied on large block sizes. This is due to the absence of signalling overhead in lower blocks, which do not require an additional parameter per PU to be sent to the decoder. This not only reduces the bit rate associated with the proposed solution but also the execution time associated to the encoder. As for the decoder, since the decision of using or not the spatial patterns was performed at the encoder, no additional complexity is observed. Nevertheless, the performance improvements are overall marginal.

Given the very small gains observed in Table 5-B, it was decided to abandon further research work on this method. Given the fact that the spatial patterns used in this technique were computed based on the same sequences that were used in the performance assessment, higher gains were expected with the tested mechanism. The observed low gains therefore do not encourage further investigation since even lower performance is expected when testing the proposed solution with arbitrary test sequences that were not used to generate the patterns. Nevertheless, studying Intra prediction in HEVC and observing the characteristics of the absolute residuals provided useful expertise in this topic. As previously mentioned, this provided valuable knowledge to develop the CIP-based improved Intra prediction technique, whose performance is reported in the next subsection.

5.4.2 Improved combined Intra prediction

This subsection reports the performance of the proposed Improved CIP schemes. The test sequences used to evaluate the performance of the techniques and the adopted test conditions are the ones used during the development of the HEVC standard [78]. Similarly to the case of the previous Intra prediction method, the tests were performed using the all Intra configuration, where all frames are Intra coded. All techniques presented in this section were applied only to the luma component.

Table 5-E shows a comparison between the performance of the original version of

CIP and the proposed Improved CIP. Both approaches were implemented on top of the HEVC reference software HM 16.6. The BD-rate metric [22] is used to compare the RD performance of each version with the original encoder as in HM 16.6. Table 5-E also reports the encoding and decoding times for both CIP versions with respect to HM 16.6. Table 5-F shows the percentage of times CIP is selected by the RDO process at the encoder for each block size where Intra prediction is computed.

Table 5-E: Performance of Improved CIP

	Improved CIP			**Original CIP [86]**		
	Y	Cb	Cr	Y	Cb	Cr
Class A	-1.4%	-3.1%	-3.1%	-1.0%	-2.1%	-2.1%
Class B	-0.9%	-1.8%	-1.9%	-0.7%	-1.3%	-1.4%
Class C	-0.9%	-1.1%	-1.2%	-0.7%	-0.8%	-0.9%
Class D	-0.9%	-1.0%	-1.1%	-0.7%	-0.8%	-0.8%
Class E	-1.2%	-2.0%	-2.0%	-0.9%	-1.5%	-1.5%
Overall	-1.1%	-1.8%	-1.8%	-0.8%	-1.3%	-1.3%
Enc. Time		218%			220%	
Dec. Time		101%			100%	

Table 5-F: Usage of Improved CIP

Block Size	Improved CIP usage
4×4	33%
8×8	30%
16×16	24%
32×32	14%

From Table 5-E, it can be observed that Improved CIP can bring benefits with respect to the original CIP implementation. Also, these tests are interesting in highlighting the performance of CIP in general, when implemented on top of recent HEVC implementations. In fact, when CIP was first proposed in [86], HEVC was still in early stages of development. Many Intra coding tools were missing from such implementation, and in general the encoder was improved by means of a variety of optimisations. This means that the performance of the original CIP technique in [86] is lower than originally presented. Conversely, Improved CIP provides consistently better RD performances than the method proposed in [86]. It is worth noting that the method achieves higher RD performances for higher spatial resolutions (class A), which might be an advantage given

the higher resolutions expected in future video formats. It can also be noticed that, even though Improved CIP is only applied for the luma component, higher gains in the chroma components are achieved. This is because the BD-rate is computed using the overall bit rate (luma and chroma) and the PSNR of each component separately. As Improved CIP brings reductions in the luma bit rate, the overall bit rate is lowered and the PSNR of the chroma components remains similar, resulting in higher chroma BD-rate values. Finally, it can be noticed that the Improved CIP method does not add any complexity with respect to the original CIP. Both solutions require around double the encoding time than HM 16.6, due to the fact that each CU needs to be tested twice (with and without using CIP). The approach has negligible effects on the decoder complexity.

Other improved RD techniques might be used to reduce the complexity associated with CIP at the encoder. Furthermore, Intra coding is associated with lower complexity than Inter coding in HEVC, which means that the increase in encoding complexity associated with CIP for an all Intra configuration has minimal impact in case Inter coding is also used. This can be observed in Table 5-G, where the performance of the proposed Improved CIP approach is reported under Random Access conditions. These results also show that Improved CIP is able to provide some bit rate savings even in a Random Access configuration, where the great majority of the video frames are Inter-coded.

Table 5-G: Improved CIP in Random Access configuration

	Improved CIP		
	Y	Cb	Cr
Class A	-0.3%	-1.0%	-0.7%
Class B	-0.3%	-0.5%	-0.6%
Class C	-0.3%	-0.4%	-0.5%
Class D	-0.2%	-0.4%	-0.2%
Overall	-0.3%	-0.6%	-0.5%
Enc. Time		108%	
Dec. Time		100%	

In order to compare the performance of CIP with other Intra prediction improvement techniques that go beyond HEVC, Table 5-H shows the BD-rate results for MPI [97]

with respect to HM 16.6. Table 5-H also shows the alternative approach proposed in Subsection 5.3.3 that combines Improved CIP and MPI.

Table 5-H: Performance of Improved CIP + MPI

	Improved CIP + MPI			**MPI [97]**		
	Y	Cb	Cr	Y	Cb	Cr
Class A	-1.9%	-2.5%	-2.5%	-1.6%	-1.5%	-1.5%
Class B	-1.3%	-1.5%	-1.7%	-1.1%	-0.8%	-1.0%
Class C	-1.0%	-1.0%	-1.1%	-0.8%	-0.5%	-0.6%
Class D	-1.0%	-0.9%	-1.0%	-0.7%	-0.4%	-0.5%
Class E	-1.5%	-1.8%	-1.7%	-1.1%	-0.8%	-0.8%
Overall	-1.3%	-1.5%	-1.6%	-1.1%	-0.8%	-0.9%
Enc. Time		218%			216%	
Dec. Time		99%			104%	

It can be observed from Tables 5-H and 5-E that the RD performance of MPI for the luma component is similar to the proposed CIP version, while CIP shows considerably better performance for the chroma components due to the overall bit rate reduction explained previously in this section. Also, the proposed combination of Improved CIP + MPI shows a consistently higher performance with respect to the CIP and MPI techniques alone. This increase in performance is achieved without additional encoder complexity. Further BD-rate gains can be achieved if both MPI and CIP are tested for each block size and two bits are used per CU to signal between the 3 options (normal HEVC Intra prediction, MPI, or Improved CIP). However, this approach involves an encoder complexity increase of about 3 times compared to HM 16.6, since each CU has to be tested 3 times at the encoder side.

5.5 Conclusion

This chapter presented two different approaches to improve the performance of Intra prediction in HEVC. The first one, derived from a preliminary study on the characteristics of Intra prediction residuals in HEVC, relies on applying predetermined spatial patterns to compensate for the inaccuracies of Intra prediction blocks. The set of spatial

patterns is known both at the encoder and decoder and different patterns are considered for different prediction modes and different block sizes. The performance of this technique shows very small gains with respect to the HEVC reference software, even when the training set of video sequences used to generate the patterns is the same as the set used to test the method. However, studying this approach provided valuable knowledge on the characteristics of Intra coding residuals, useful to optimise the second approach proposed in this chapter.

The second approach proposed in this chapter is based on the CIP mechanism, which uses information from within the block being predicted to improve Intra prediction in HEVC. An improvement technique was proposed to avoid mismatches in the CIP values computed at the encoder and decoder at the top and left borders of the prediction blocks. The performance results show that the Improved CIP is able to achieve consistent BD-rate gains for all classes of test sequences, and in particular an average of 1.4% bit rate reduction for class A sequences. A combination of Improved CIP with another Intra prediction improvement technique, MPI, was also proposed. The performance results show that this Improved CIP + MPI can achieve an average BD-rate gain of 1.9% for class A while maintaining the same encoding complexity of both Improved CIP or MPI. Higher gains could be achieved if both solutions were fully combined, at the cost of increased complexity (around 3 times higher than the reference software).

Chapter 6

Encoding time control for practical HEVC encoding applications

As mentioned in Chapter 2, video compression standards like HEVC only define the syntax and semantics of the encoded bit stream and the operation of the decoder. Therefore, there is substantial flexibility in the operation of video encoders, as they have the freedom to choose the best way to use the tools provided by the standard. In other words, the operation of an encoder can be tuned to achieve the desired performance for the target application, not only in terms of compression efficiency, but also in terms of encoding time, memory consumption, frequency of random access points in the stream, fixed bit rate restrictions and so on.

In this context, the focus of this chapter is on optimising HEVC for practical encoding use cases, where controlling the encoding time of video encoders is essential. In particular, in some applications, video content needs to be encoded and uploaded to a remote destination within a pre defined amount of time. In order to guarantee that the overall processing time does not exceed certain time constraints, a system that performs joint

encoding and uploading time control is needed. Such a system requires the flexibility to control both the time spent in the encoding process and the bit rate. This is because the latter significantly influences the transmission time, especially for transmissions under low bandwidth constraints.

This chapter proposes a novel approach to address this challenge by adapting the QP of a video encoder in order to meet overall processing time requirements, including both encoding and uploading time. The proposed QP adaptation approach relies on mechanisms to accurately predict the encoding time and bit rate during the encoding process for the incoming group of pictures. This in turn allows adequate QP selections that result in accurately meeting the overall time constraints.

Overall, the main contributions proposed in this chapter are:

- An algorithm for joint encoding and uploading time control based on adaptive QP selection for groups of frames.
- A technique to accurately estimate encoding time variations with the QPs used for encoding a group of frames, based on the percentage of non zero quantised coefficients, ρ.
- A technique to reuse ρ estimations to accurately estimate bit rates for a group of frames, based on a piecewise interpolation/extrapolation scheme.

The rest of this chapter is organised as follows. Section 6.1 gives a more detailed explanation of the motivation and the application scenarios targeted by the proposed system. Section 6.2 provides a brief overview of the background work relevant to the techniques proposed in this chapter. Section 6.3 describes the overall joint encoding and uploading time control, including the details of the proposed encoding time and bit rate estimation methods. Finally, Section 6.4 provides experimental results that evaluate the performance of the proposed method.

6.1 Motivation

In off-line contribution scenarios where the content needs to be transmitted after compression, both the encoding time and the expected produced bit rate need to be jointly controlled in order to meet possible overall encoding and uploading time targets. A potential use case with such conditions may be journalists or videographers in the field contributing high quality video content to a central repository for news, documentary making or video production, for example. Contribution material is typically transmitted at high levels of quality and high resolutions, and thus cannot be transmitted in real time without very high bandwidth links dedicated to this purpose. However, in most cases, time constraints still need to be considered, imposing limitations both on the encoding time and output bit rates.

In these cases, media professionals want the video content to be not only efficiently compressed, but also uploaded to a central repository, so that it reaches the intended destination in a predefined amount of time (while still targeting the maximum possible level of quality). Frequently, such deadlines are crucial to ensure timely delivery of the edited programmes. On the other hand, when conventional video encoders encode video content with the highest possible quality, they may require very high encoding times and very high bit rates, which consequently may lead to long uploading times. Moreover, professionals are often in remote locations where the availability of bandwidth may be limited and unreliable. Media professionals are therefore forced to compromise on quality in order to meet the deadlines, while still not having any guarantee that the content will reach the destination on time.

It is important to emphasise that encoding and uploading times must be considered jointly. As previously mentioned, controlling only the encoding complexity ensures no control on uploading time, which can be significant for high bit rates. Conversely, fixing the uploading time by specifying a target bit rate gives no guarantees with respect to the encoding complexity, since this is highly dependent on the characteristics of the video

content. Rate control algorithms typically work by adaptively changing the QP used in the encoding process, so that the appropriate number of bits is spent to encode each portion of the content, in order to meet the target bit rate. These QP variations are derived depending on the content being encoded and have a non negligible impact on the encoding time, as shown in experiments reported in this chapter.

The approach proposed in this chapter adaptively changes the QP during the encoding process based on accurate estimation algorithms which are capable of predicting jointly the time necessary for encoding video content with a specific QP, as well as the resulting bit rate. The selected QP values are fixed for a given group of frames in advance. By using accurate encoding time and uploading time estimations, the encoder can select the lowest QP for each group of frames that satisfies the imposed time constraints, consequently providing the maximum possible quality. The algorithm was tested on a large set of video content, showing that it can meet the time constraints with high accuracy.

6.2 Background work

As mentioned in the previous section, the core of the framework proposed in this chapter is based on accurate bit rate and encoding time estimation techniques. In this section, a brief overview of the most relevant bit rate estimation techniques used by rate control algorithms is first presented. The second subsection gives a brief overview of the most relevant work in the literature related to encoding time control.

6.2.1 Bit rate control

Many algorithms for controlling the output rate of video codecs have been proposed in the literature, as this is an essential tool for most practical video coding applications. In general, rate control algorithms first define the number of bits that should be spent for a given block, frame or group of frames, according to the available bit budget. Given

this bit budget allocation, a decision on the best encoding parameters is then performed, based on estimations that relate these parameters with the number of bits needed to encode the content.

Most rate control algorithms attempt to model a relationship between the coded bit rate and the quantisation step used for encoding. The reference software of the AVC standard (Joint Model, JM) provides a method to model such relationship [99]. This method uses the Mean Absolute Difference (MAD) of the residuals of previously encoded basic units (which can be as small as a macroblock) to estimate the MAD of future basic units with a linear model. The estimated MAD is then used to compute the most appropriate quantisation step for a specific bit budget, using a quadratic rate-quantisation relationship, assuming the residual information follows a Laplacian distribution [100]. The resulting quantisation step is then mapped to the corresponding QP to use in the encoding process. The same quadratic rate-quantisation model was also proposed in the context of HEVC [101] and was used in the rate control algorithm of the early versions of the HEVC reference software (HM) implementation [98]. However, the quad-tree partitioning structure in HEVC described in Chapter 2 introduces significant differences in terms of the associated coding residuals with respect to AVC and therefore adjustments may be needed to apply the model in this context.

Another group of rate control algorithms relies on the relationship between rate and the Lagrangian multiplier, λ. In order to achieve high compression efficiency, most video encoders select the best coding option based on rate distortion optimisation. A cost J is computed for each coding option, typically as $J = D + \lambda \cdot R$, where D is the distortion between the original and reconstructed content when using the currently tested option, R is the corresponding rate and λ is a Lagrangian multiplier used in the optimisation process [2][102]. λ domain rate control algorithms use a model that establishes a relationship between the bit rate and λ, selecting the most suitable λ value for the desired target bit rate. This λ value is then mapped to the corresponding quantisation step. In the work in [103], a λ domain rate control algorithm is proposed

where the relationship between rate and λ is modelled with a power function. The reported results show higher accuracy than previously proposed rate-quantisation models in the context of HEVC. This rate control algorithm was adopted in the most recent version of the HEVC reference software.

Finally, another group of rate control algorithms relies on the relationship between rate and the percentage of non zero coefficients after quantisation, denoted by ρ. By using this relationship, ρ can then be mapped to a quantisation step, as in the previously described methods. One example of a relevant ρ domain rate control scheme in the literature is the method proposed in [12], where a quadratic ρ domain rate model is used in a hierarchical bit allocation scheme for rate control in an HEVC encoder. The proposed algorithm uses a linear relationship in the ρ domain between the bits associated with texture and the number of non zero transformed coefficients. The number of non zero transformed coefficients is then modelled as a quadratic function of the quantisation step. Other relevant ρ domain rate control systems include the rate shape smoothing algorithm proposed in [104] to obtain smoother rate distribution and ensure consistent picture quality in the context of an H.263 [105] encoder and the work in [106], which applies a ρ domain rate control algorithm to scalable video coding.

In this chapter, a ρ domain piecewise fitting model is proposed to model the relationship between ρ and the number of bits spent. This model is used to perform bit rate estimations for a group of frames to be encoded. The approach reuses the ρ information needed to perform encoding time estimations, as further detailed in the next section.

6.2.2 Encoding time control

Considering that most practical video compression applications operate under encoding time constraints, mechanisms for encoding time reduction/control are essential. Even though complexity increase is evaluated and taken into account during the development of new video coding standards, providing significantly higher compression performance

typically comes at the cost of significantly higher encoder complexity, as in the case of HEVC [107]. For this reason, many complexity reduction techniques for video encoding have been studied for different video coding standards. Such techniques should ideally have low impact on rate distortion performance. In the case of HEVC, these techniques typically rely, for example, on fast algorithms to select block partitions which are close to optimal in a rate distortion sense, without having to perform exhaustive searches. These include techniques such as fast CU splitting decisions based on the fly statistics and available intermediate encoding data for both Intra [108] and Inter coding [109] or fast CU splitting using decision trees obtained through data mining techniques [110][111]. Similarly, fast decisions for selecting prediction modes [112] or fast algorithms to perform motion estimation [113][114] were also proposed, all with the purpose of reducing encoding execution time by tackling different aspects of the overall encoding process.

All these techniques provide efficient speed-ups for the encoding process. However, some applications benefit from having higher flexibility to control the encoder complexity in order to maximise the rate distortion performance of video encoders, depending on the specific execution time constraints imposed by the application. This type of rate-distortion-complexity optimisation approach has been less explored in the literature. Assuming the encoding time as a measure of complexity, an algorithm for rate-distortion-complexity control is proposed in [115] by defining a complexity budget allocation scheme and using adaptive search algorithms to efficiently use the assigned computational budget at a macroblock level. The technique was proposed in the context of a practical implementation of the AVC standard [116]. As for HEVC, the method in [117] proposes a combination of medium and fine granularity encoding time control algorithms to keep the encoding time below a predefined target for each group of pictures. This algorithm controls the encoding operation by switching through sets of complexity reduction techniques identified through rate-distortion-complexity analysis [118].

The encoding time control methods mentioned in the previous paragraph tackle the problem of complexity adaptation by modifying the encoder operation in order to perform

more or less extensive searches. In the context of this chapter, where restrictions are considered not only to the encoding time but also to the time needed for transmitting the resulting encoded bit stream, the bit rate of the encoded video also has an important role in the overall time control mechanism. As an example, even very fast configurations of the encoder may fail to reach a given joint encoding and uploading time target if they operate at high bit rate points, which lead to high uploading times under low bandwidth network conditions. For this reason, bit rate and time control methods need to be combined together to meet the overall time constraints. Furthermore, in all the methods mentioned in this section, the variations of the encoding time with the QP used for encoding are not taken into consideration. In [119], it is reported that for both AVC and HEVC reference software implementations, encoding the same sequence with different QPs can lead to an encoding time increase of almost 30%. Similar conclusions can be drawn from some experiments reported in this chapter for the case of a practical HEVC video encoder. This difference in encoding time with different QPs is especially important for applications targeting high quality compressed video for which encoders can produce very high bit rate compressed bit streams.

Given the lack of solutions available in the literature that address the joint encoding and uploading time constraints referred in the previous paragraph, this chapter proposes a joint encoding and uploading time control by combining both bit rate and encoding time control techniques, taking into account the relevant influence of QP variations on the encoding time.

6.3 Time and rate constrained encoding

Differently from the schemes described in the previous section, the algorithm presented here aims to control both the encoding time and the produced rate, so that an overall target time constraint can be satisfied, including encoding and uploading times. A simplified scheme of the type of use case addressed by the proposed method is shown in

Figure 6.1.

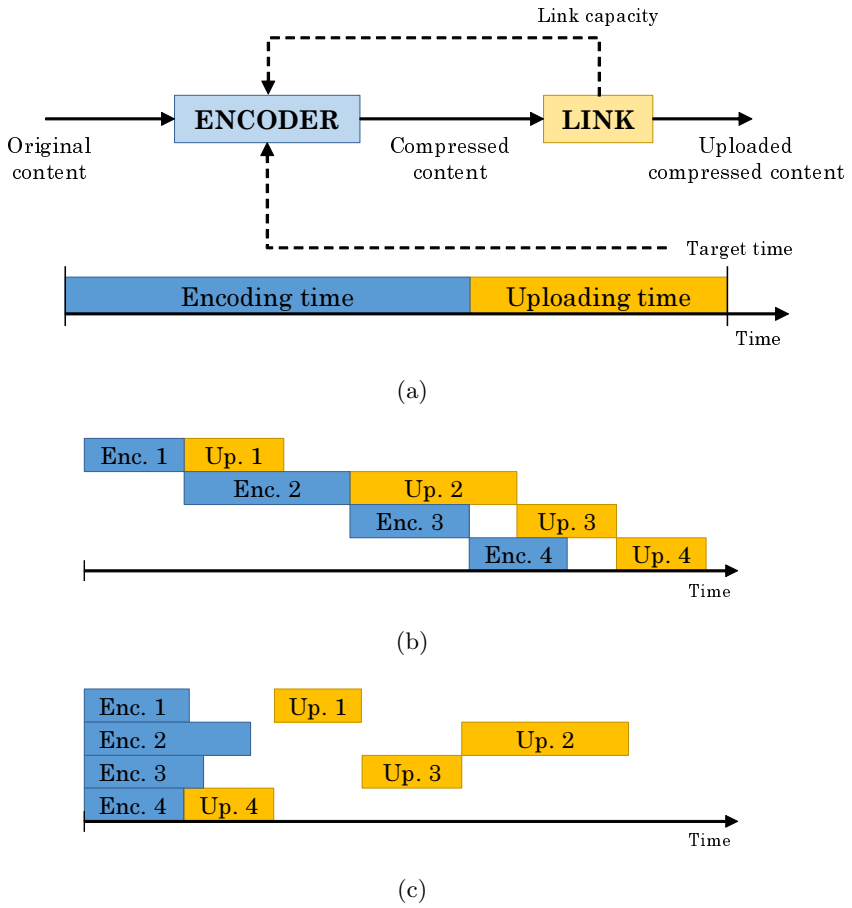


Figure 6.1: General encoding and uploading scheme. a) Sequential processing (considered in the proposed method for simplicity); b) Content encoded in chunks and uploading performed in parallel; c) Encoding of chunks performed in parallel and compressed chunks are uploaded in arbitrary order.

It is important to note that, for simplicity, encoding and uploading are considered as sequential processes in the rest of this chapter, as illustrated in Figure 6.1(a). In practical applications, in general, the input video sequence is segmented into chunks, typically of a few seconds in length. Uploading of a chunk can begin as soon as it has been encoded, while the encoding process of the next chunk can be performed in parallel, as illustrated in Figure 6.1(b). Figure 6.1(c) illustrates an additional scenario where the encoding process of each chunk can be performed in parallel, in case this is supported by

the computational resources available for encoding. In all these situations, a good joint control of the encoding time and uploading time is beneficial since the uploading time can have non negligible influence on the overall delivery time in each of these scenarios. Without loss of generality, the proposed method described in the rest of this section was designed for the situation depicted in Figure 6.1(a). The method can though be easily adapted to account for any possible variation involving the parallel processing of video chunks like the ones illustrated in Figure 6.1(b) and 6.1(c). This is because the estimations of the encoding time and total number of bits spent (the two main technical challenges tackled by the proposed approach) are independent of the adopted chunk parallelisation scheme.

Video compression standards like AVC and HEVC are based on the so called hybrid video coding approach. Hybrid video encoders typically evaluate the encoding tools available in the standard and select the best options, typically based on a trade-off between encoded bit rate and distortion. Each of these decisions influence the rest of the encoding loop, resulting in very different encoding times. For this reason, it is typically challenging to predict how long the encoder will take to encode a given piece of content. Moreover, depending on the type of content to encode and the output quality targeted by a given application, the encoding process can generate compressed bit streams with very different bit rates, as shown in experiments reported in Section 6.4.

The proposed algorithm, described in the next subsection, was designed assuming a typical hierarchical frame coding structure based on Structures of Pictures (SOP) layers. SOPs define parameters such as encoding order (which may be different from display order, referred to as Picture Order Count, POC), the reference frames, QP offset and so on. Without loss of generality, the SOP structure depicted in Figure 6.2 was periodically used to encode the sequence, in accordance with the Random Access configuration defined in [78]. The proposed algorithm is based on varying the base QP on a SOP basis (and consequently the corresponding QPs for each frame according to the QP offsets), so that information extracted from frames belonging to each SOP layer

can be used to predict the encoding time, as well as the number of bits spent in the next SOP.

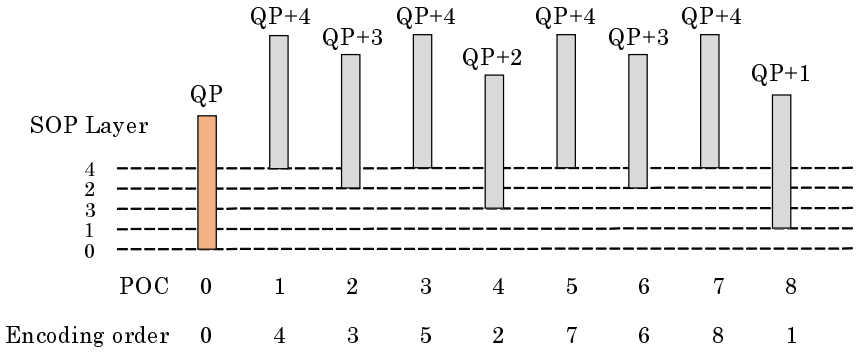


Figure 6.2: Hierarchical layers of a SOP in RA configuration.

6.3.1 General algorithm

Denote as $\overline{T}_{tot}$ the total target time available from the moment the encoding process is triggered to the moment the content must reach the destination. For the sake of simplicity, assume that uploading happens under ideal conditions within a network channel with a fixed known available bandwidth equal to W_{link}. In practice, this bandwidth may vary with time, with no impact on the workflow of the proposed approach. Finally, assume that the considered sequence is composed of a total number of N SOPs and each SOP is referred to as SOP n, with $n \in \{0, \ldots, N-1\}$. It is important to note that the length of the video sequence to encode is always known.

The proposed approach operates by assigning a specific target time to each SOP. Before starting encoding the first SOP in the sequence, a uniform distribution of the total time is assumed among all SOPs. Formally, the first SOP, referred to as SOP 0, is initially assigned a target time $\overline{T}_{tot,0} = \overline{T}_{tot}/N$. A predefined initial QP value, denoted as q_0, is used to encode this SOP. The same value of q_0 is used regardless of the specific time constraints, as no prior information is available before starting the encoding process. Additional methods to select q_0 can be further investigated to optimise

the proposed algorithm. It should be noticed, however, that the impact of the initial value q_0 is relatively marginal to the performance of the algorithm when encoding long video sequences, as the QP will quickly adapt to the specified time constraints.

The first SOP is encoded using q_0 as base QP. When the encoding process of this SOP is complete, a total number of bits B_0 is produced in an encoding time $T_{enc,0}$. During the encoding process, relevant information is collected (as further detailed in Subsections 6.3.2 and 6.3.3) so that the encoder can perform decisions on the QP to use on the next SOP. Considering the transmission of the encoded bit stream, the total time necessary to encode and upload the first SOP, $T_{tot,0}$, is obtained as:

$$T_{tot,0} = T_{enc,0} + \frac{B_0}{W_{link}} \tag{6.1}$$

The target time for the next SOP, $\overline{T}_{tot,1}$, can then be refined by computing how much time is left (with respect to the total target $\overline{T}_{tot}$) using the actual time spent on the first SOP. Using information extracted while encoding the first SOP, the methods described in Subsections 6.3.2 and 6.3.3 provide accurate estimations of the encoding time and total number of bits that will be obtained if a different QP value is used to encode the next SOP. Based on these estimations and on $\overline{T}_{tot,1}$, the encoder then selects a new QP value q_1 to use in SOP 1, so that the total time for encoding and uploading this SOP is as close as possible to the target. Since it is undesirable to introduce abrupt variations of QP from SOP to SOP, QPs are limited to a variation of ± 5 between consecutive SOPs. This process briefly described for SOP 1 can be generalised according to the following algorithm:

1. Before encoding a given SOP n, the target time for the SOP is computed taking into account the total target time as well as the actual time spent for the previously processed SOPs, or formally:

$$\overline{T}_{tot,n} = \frac{\overline{T}_{tot} - \sum_{i=0}^{n-1} T_{tot,i}}{N - n} \tag{6.2}$$

2. A set of QP values is considered, namely $\mathbf{Q} = \{q_{n-1} - 5, q_{n-1} - 4, \ldots, q_{n-1} + 5\}$. For each valid q in $\mathbf{Q}$, the following steps are performed:

   (a) Using information extracted from SOP $n-1$, an estimation of the encoding time to encode SOP n using q is computed as $\tilde{T}_{enc,n}(q)$ (described in Subsection 6.3.2).

   (b) Using information extracted from SOP $n-1$, an estimation of the number of bits necessary to encode SOP n using q is computed as $\tilde{B}_n(q)$ (described in Subsection 6.3.3).

   (c) A prediction of the total time necessary for encoding and uploading with q is finally computed as:

$$\tilde{T}_{tot,n}(q) = \tilde{T}_{enc,n}(q) + \frac{\tilde{B}_n(q)}{W_{link}} \tag{6.3}$$

3. The QP value q_n to use for SOP n is selected as the minimum value that satisfies the following constraint, in order to maximise quality:

$$\min\{q \colon q \in \mathbf{Q}, \tilde{T}_{tot,n}(q) \leq \overline{T}_{tot,n}\} \tag{6.4}$$

4. Finally, SOP n is encoded using q_n as base QP. The actual encoding time $T_{enc,n}$ as well as number of bits spent on the SOP, B_n, are computed and the actual total time for this SOP is obtained as:

$$T_{tot,n} = T_{enc,n} + \frac{B_n}{W_{link}} \tag{6.5}$$

This algorithm relies on two specific techniques for computing an estimation of the encoding time and bit rate on a SOP level for a given QP value (as in steps 2.a and 2.b). Both the encoding time and number of bits estimations are based on the ratio of non

zero coefficients obtained after quantisation over the total number coefficients, hereafter denoted as ρ. The following subsections present details of such estimation techniques.

6.3.2 SOP-level encoding time estimation

The general time control algorithm in Subsection 6.3.1 assumes that the encoder is capable of accurately estimating the time it would take to encode the next SOP in the sequence for specific values of the QP. A technique to obtain such estimation is described in this subsection, based on information extracted during the encoding process.

By analysing the workflow of the typical residual coding operations depicted in Figure 6.3, it can be concluded that the transform process is not affected by the QP value and the same happens with the corresponding time spent by the encoder performing it. As for quantisation, even though the QP value determines the quantisation step to use, the actual time spent in quantisation is not affected by the quantisation step because all coefficients in the TB need to be processed. Finally, the resulting quantised coefficients, also denoted as coefficient levels, are entropy encoded in HEVC with CABAC [49]. Differently from the previous two operations, the time spent in the overall entropy encoding process of the quantised coefficients varies significantly with the QP used for quantisation. As the QP increases, the quantisation step used in the quantisation process also increases, and more coefficients in the TB are quantised to 0, meaning that the entropy encoding operation needs to process a lower number of non zero coefficient levels. For lower QPs, more non zero quantised coefficients will be fed to the entropy encoder, leading to higher encoding times. The entropy encoding process is therefore the main factor responsible for encoding time variations on a given TB when using different QPs, where such variations are directly correlated with the ratio of non zero levels over the total number of coefficients in the TB. For this reason, the first step for obtaining a reliable estimation of the encoding time for a given QP is to compute the value of ρ obtained with such QP value on a TB.

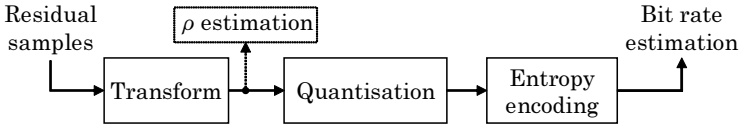


Figure 6.3: Estimation of ρ in the context of the main operations performed during the residual coding process in a typical HEVC encoder.

The estimation of ρ in the proposed method is performed before the quantisation process, as illustrated in Figure 6.3. Conceptually, the scalar quantisation process in a typical HEVC encoder can be described by Eq. (2.9) (see Chapter 2). In Eq. (2.9), the absolute values of a given coefficient $c_{i,j}$ that will result in coefficient levels different from zero (i.e. the corresponding level, $v_{i,j}$, will have an absolute value higher than or equal to 1) needs to satisfy the following condition:

$$|c_{i,j}| \geq (1 - d) \cdot \delta_q. \tag{6.6}$$

Using the condition from Eq. (6.6), for a given TB of size $A \times A$, the ratio of non zero levels, ψ, obtained with a given q can be computed as:

$$\psi(q) = \frac{k_q}{A \cdot A} \tag{6.7}$$

where k_q is the number of coefficients $c_{i,j}$ that satisfy Eq. (6.6) when quantised with q.

In practice, the scalar quantisation process in Eq. (2.9) is defined in HEVC based on equivalent scaling and shifting operations according to the used quantisation step δ_q [47]. Different scaling and shifting factors are defined according to the QP, slice type (Intra or Inter), colour component and transform size. Therefore, in terms of practical implementation of the method described in this chapter, all possible threshold values corresponding to the right part of Eq. (6.6) are pre computed based on the corresponding scaling and shifting factors defined in HEVC and stored in a look up table, to reduce the time needed for the computation of ψ for a given range of QPs. Furthermore, the

number of comparisons that needs to be performed for each coefficient is lower than the number of QPs in the range, since if a coefficient $c_{i,j}$ is not large enough to be quantised to a non zero value for a given QP, it will also be quantised to 0 for any higher QP. This significantly reduces the number of comparisons necessary for each TB, reducing the impact of the computation of ψ in the overall encoding time.

The process described at TB level is used at the frame level to provide an estimation of the ratio of non zero coefficient levels over the total for a given frame. In particular, denote as M the total number of TBs tested within the current frame. Denote as $k_{q,m}$ the number of coefficients $c_{i,j}$ that satisfy Eq. (6.6) for a QP value of q on TB m of size $A_m \times A_m$. Then the non zero level ratio at the frame level, $\rho(q)$, is given by:

$$\rho(q) = \frac{\sum_{m=0}^{M-1} k_{q,m}}{\sum_{m=0}^{M-1} A_m^2}. \tag{6.8}$$

It is important to note that, when Eq. (6.8) is used while encoding a frame using an actual QP value different from q, the obtained $\rho(q)$ is only an estimate of the average ratios of non zero coefficient levels. This is because using different QP values produces different reconstruction samples, which are then used as reference samples for computing the prediction in subsequent blocks. Hence, this difference is propagated resulting in different residuals which, when input to the process in Figure 6.3, have an impact on the number of non-zero coefficient levels, and consequently on the reliability of the estimation computed with Eq. (6.8).

In order to evaluate how much such differences would impact the reliability of $\rho(q)$, the sequence *Manege* was encoded using fixed QP values ranging from 11 to 45 (see Section 6.4 for sequence details). The actual ratio of non zero levels over the total was then computed for each frame in SOP layer 1, and averaged over the total number of frames in this SOP layer, for each QP value. These were then plotted in Figure 6.4, represented as dots in the plot. Then, the encoding process was performed using a QP value of 18 and the estimates $\rho(q)$ were computed with Eq. (6.8) and plotted as a solid

line in the figure. It can be seen that the estimation is accurate, especially for QPs close to the actual QP being used. A similar behaviour was obtained for other sequences and SOP layers. As the QP variation is limited to ±5 between consecutive SOPs, $\rho(q)$ was used as an estimate of the actual average ratio of non zero levels for the remaining steps of the proposed techniques.

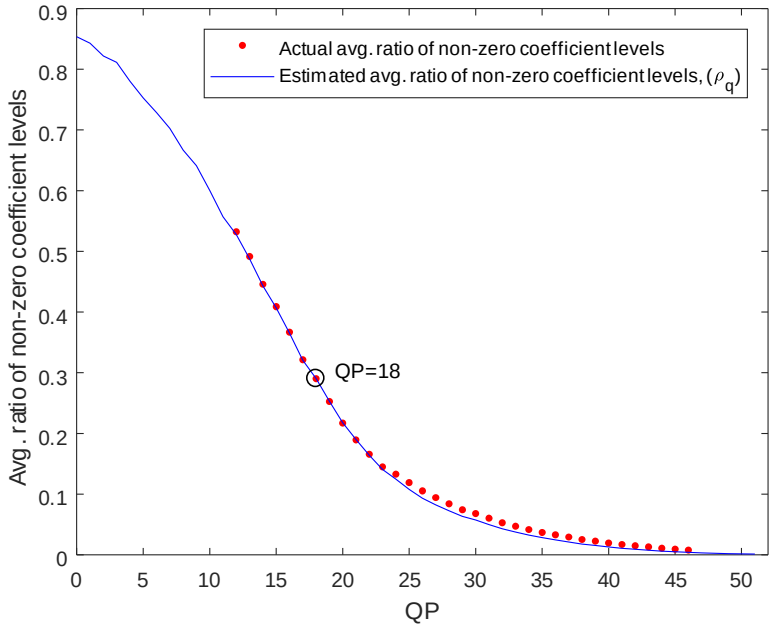


Figure 6.4: Actual and estimated average ratios of non zero coefficient levels for SOP layer 1 in the *Manege* sequence.

The actual average ratio of non zero levels is highly correlated with the time necessary to perform entropy coding of the quantised coefficients on a given frame. To illustrate such relationship, some experiments were performed using again the sequence *Manege*, encoded with a constant QP ranging from 11 to 45. A specific frame (frame 8) was considered as an example, where for each encode the actual average ratio of non zero levels was computed. Also, the encoder was modified to keep track of how much time is required to perform entropy coding of the quantised coefficients for that specific frame, denoted as $t_{EC}(q)$. The plot in Figure 6.5 shows the results of such tests, where $t_{EC}(q)$ is plotted against the actual average ratio of non zero levels for QP values ranging from 11 to 45. From Figure 6.5, modelling the relationship between the ratio of non zero levels

and the time spent on entropy encoding with a linear approximation was considered to be sufficiently accurate. Similar behaviour was observed for other frames in other sequences and other SOP layers. Therefore, a linear model was used to estimate the entropy encoding time of the quantised coefficients, which is then used to estimate the encoding time, as explained in the following.

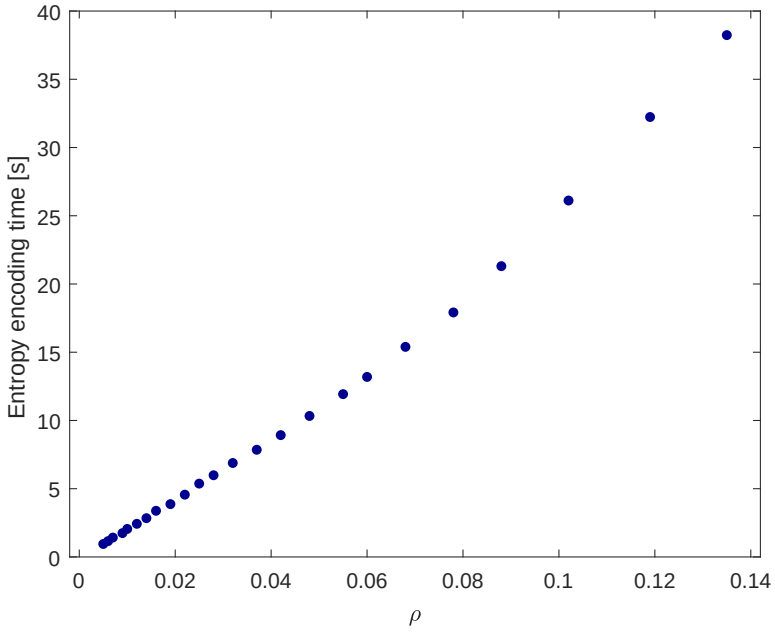


Figure 6.5: Entropy coding time versus average ratio of non zero coefficient levels for frame 8 of the *Manege* sequence, with QP values from 11 to 45.

Experiments reported in Section 6.4 show that different SOP layers behave differently with respect to encoding times and for that reason the proposed method is applied independently to each SOP layer. Based on features extracted from frames in a given SOP layer, an estimate of the encoding time that would be obtained if these frames were encoded with different QPs is first computed. Then, this is used to estimate the total encoding time for the next SOP for a range of different QP values. The estimation of the encoding time is based on the previously described computation of $\rho(q)$.

Assume that a given frame at a given SOP layer, l, in a given SOP n is being encoded and denote as $q_{n,l}$ the QP value being used to encode this frame (obtained as base QP

used in SOP n, q_n, plus the QP offset corresponding to the SOP layer l). Denote as $t_{EC}(q_{n,l})$ the total time necessary to entropy encode the quantised transform coefficients in the frame. Similarly, denote as $t_{rem}(q_{n,l})$ the total remaining time necessary for encoding the frame (measured from the instant the frame starts encoding, to the instant the last bit is written in the bit stream). Denote as $t_{enc}(q_{n,l}) = t_{EC}(q_{n,l}) + t_{rem}(q_{n,l})$ the total encoding time of the frame. For each QP value q in $\mathbf{Q} = \{q_{n,l} - 5, q_{n,l} - 4, ..., q_{n,l} + 5\}$, the encoder can compute $\rho(q)$ using Eq. (6.8). Finally, the encoder also computes the actual average ratio of non zero levels over the total obtained while encoding with $q_{n,l}$, denoted as $\rho(q_{n,l})$. Given the assumption that t_{EC} and ρ are linearly correlated, for a given value of q, the following can be computed:

$$\tilde{t}_{EC}(q) = \frac{t_{EC}(q_{n,l})}{\rho(q_{n,l})} \cdot \rho(q), \tag{6.9}$$

where $\tilde{t}_{EC}(q)$ is the estimated time for entropy encoding the quantised coefficients in the frame obtained with a QP value of q. Finally, the total estimated encoding time for the frame when encoded with a QP value q can be computed as:

$$\tilde{t}_{enc}(q) = \tilde{t}_{EC}(q) + t_{rem}(q_{n,l}) \tag{6.10}$$

This process performed at the frame level is then used to perform encoding time estimations at a SOP level. Denote as L the number of SOP layers in a SOP, which is assumed to be 4 according to the adopted configuration (as in Figure 6.2). Denote as F_l the number of frames in each layer l (for instance, $F_3 = 2$). Using Eq. (6.10), it is possible to have an estimation of the total encoding time for each frame in the SOP, for each allowed QP value in the considered range $\mathbf{Q}$. Denote as $t_{enc,l}(q)$ the average total estimated encoding time computed for all frames in the SOP belonging to SOP layer l. Finally, the total estimated encoding time for the whole SOP encoded with a QP value of q can be computed as:

$$\tilde{T}_{enc,n}(q) = \sum_{l=0}^{L-1} F_l \cdot t_{enc,l}(q). \tag{6.11}$$

The estimated time obtained with Eq. (6.11) can then be used in step 2.a. in the algorithm presented in Subsection 6.3.1.

6.3.3 SOP-level bits prediction

In addition to the encoding time estimation of the next SOP for different QPs, another essential element of the proposed algorithm is the estimation of the number of bits necessary to encode SOP n using a QP value of q, denoted as $\tilde{B}_n(q)$ in step 2.b in the general algorithm described in Subsection 6.3.1.

The number of bits necessary to encode a given frame is related to the ratio of non zero levels left after quantisation. As such, a model is proposed to estimate these bits based on the estimated average ratio of non zero levels when encoding with q, previously denoted as $\rho(q)$. Similarly to the approach in the previous subsection, this estimation is also based on the continuous refinement of the model from real observations obtained while encoding. While the approach for encoding time estimation considered only a single observation (the most recent), a more complex model is adopted in the estimation of the number of bits. As explained in the rest of this subsection, this estimation takes into consideration a number of pairs of $\rho(q)$ and the corresponding number of bits necessary to encode a frame, obtained from previously encoded frames. The method is applied independently to frames from different SOP layers.

Let $b(q)$ denote the total number of bits needed to encode a given frame with a given QP value of q in SOP n. Considering the process described in Subsection 6.3.2, $\rho(q)$ is available for all QPs in the allowed range $\mathbf{Q} = \{q_{n,l} - 5, q_{n,l} - 4, ..., q_{n,l} + 5\}$, along with the real ratio of non zero levels $\rho(q_{n,l})$. Therefore, after encoding a frame in SOP layer l, the pair of real observations $\{b(q_{n,l}),\ \rho(q_{n,l})\}$ is available together with the estimated

$\rho\,(q)$.

A power function of the following type is used to model the relationship between $\rho\,(q)$ and the estimated number of bits necessary to encode the frame, $\tilde{b}\,(q)$:

$$\tilde{b}\,(q) = \alpha \cdot \rho_q^{\beta}. \tag{6.12}$$

In Eq. (6.12), α and β are fitting parameters adjusted according to past observations. In particular, a number of G previous SOPs are considered to adjust these parameters. Without loss of generality, G is set to 6, which corresponds to approximately 1 second of video at a frame rate of 50 fps for a fixed SOP size of 8.

The following is then considered. It is assumed that there are a total of S stored pairs, corresponding to pairs extracted from all frames in the previous G SOPs, belonging to the current SOP layer l. Denote these as $\{\{b_0, \rho_0\}, \{b_1, \rho_1\}, ..., \{b_{S-1}, \rho_{S-1}\}\}$. For a given QP value of q in the interest range $\mathbf{Q}$:

- If $\rho\,(q)$ is lower than all ρ values in the stored pairs, a linear model is used with $\beta = 1$. The minimum estimated ratio among all stored values ρ_{min} is extracted, along with the corresponding b_{min}. Finally, $\alpha = b_{min}/\rho_{min}$.

- If $\rho\,(q)$ is higher than all ρ values in the stored pairs, a linear model is used with $\beta = 1$. The maximum estimated ratio among all stored values ρ_{max} is extracted, along with the corresponding b_{max}. Finally, $\alpha = b_{max}/\rho_{max}$.

- If there are at least two pairs $\{b_x, \rho_x\}, \{b_y, \rho_y\}$ in the stored set such that $\rho_x < \rho\,(q) < \rho_y$, the fitting parameters are defined so that they satisfy:

$$\begin{cases} b_x = \alpha \cdot \rho_x^{\beta} \\ b_y = \alpha \cdot \rho_y^{\beta} \end{cases} \tag{6.13}$$

Hence, in this case, the values of α and β are given by

$$\begin{cases} \beta = \frac{\log(b_x)-\log(b_y)}{\log(\rho_x)-\log(\rho_y)} \\ \alpha = \frac{b_y}{\rho_y^{\beta}} \end{cases} \tag{6.14}$$

When predicting frames in the second SOP of the video sequence, it may occur that only one pair is available in the set for some SOP layers. In this particular case, the value of β is set to 1, meaning that Eq. (6.12) becomes a linear model and the value of α is given by $\alpha = b_0/\rho_0$. Figure 6.6 illustrates the interpolation/extrapolation types used for the three different scenarios of the fitting parameters computation.

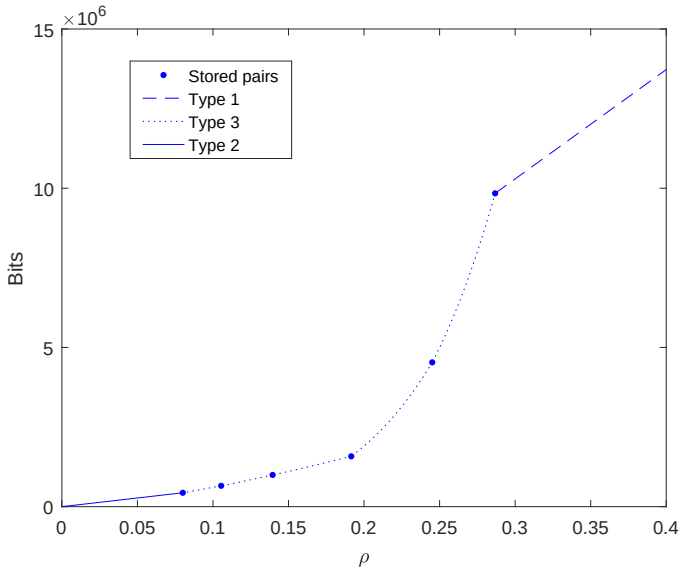


Figure 6.6: Visualisation of the proposed 3 interpolation/extrapolation types using 6 stored pairs of past observations.

Finally, denote again as F_l the number of frames in each layer l. Using the previous equations, it is possible to have an estimation of the total number of bits necessary to encode each frame in the SOP, for each allowed QP value in the considered range. The average number of bits for all frames in a given SOP layer l is then computed, denoted as $\tilde{b}_l(q)$. The total estimated number of bits to encode the whole SOP n with a QP value of q can be computed as:

$$\tilde{B}_n(q) = \sum_{l=0}^{L-1} F_l \cdot \tilde{b}_l(q). \tag{6.15}$$

The estimated number of bits obtained with 6.15 can then be used in step 2.b. in the algorithm presented in Subsection 6.3.1.

6.4 Performance evaluation

The proposed time control method was tested to evaluate its ability to accurately meet the conditions imposed by different time and bandwidth constraints. In this section, the conditions under which these tests were performed are first described. Preliminary experiments that motivated the main design choices are then reported, followed by the detailed assessment of the performance of the proposed time control method, including the accuracy of the proposed intermediate estimations and the performance of the overall framework.

6.4.1 Test conditions

The proposed method was evaluated under a variety of test conditions, selected to verify its effectiveness in various possible use cases. The test material includes HD content with spatial resolutions of 1280×720 and 1920×1080 and UHD content with a spatial resolution of 3840×2160. All selected test sequences have a duration of 10 seconds, regardless of their temporal resolution, which can be 24, 50 or 60 fps. Table 6-A shows the selected test sequences and the respective temporal and spatial resolutions. All sequences are either publicly available or belong to the JCT-VC common test conditions [78].

In order to simulate different encoding and uploading scenarios, different bandwidths and total encoding times were used in the reported experiments. To simulate challeng-

Table 6-A: Selected test sequences, resolutions and target times.

Resolution	Selected sequences	Target times (encoding + uploading) [s]
3840 × 2160 @ 50 fps	NingyoPompoms, Somersault, ParkAndBuildings	6000, 7000, 8000, 9000, 10000, 11000
3840 × 2160 @ 60 fps	Manege, Sedof, ShowDrummer	6000, 7000, 8000, 9000, 10000, 11000
1920 × 1080 @ 24 fps	ParkScene, Kimono, RushHour	650, 800, 950, 1100, 1250, 1500
1920 × 1080 @ 50 fps	BasketballDrive, Cactus, ParkDancers	1000, 1300, 1600, 1900, 2200, 2500
1920 × 1080 @ 60 fps	BQTerrace	1000, 1300, 1600, 1900, 2200, 2500
1280 × 720 @ 50 fps	DucksTakeOff, Stockholm, ParkJoy	500, 750, 1000, 1250, 1500, 1750
1280 × 720 @ 60 fps	FourPeople, Johnny, KristenAndSara	500, 750, 1000, 1250, 1500, 1750

ing conditions, bandwidths of 128, 256 and 512 kbps were considered. Relevant total target times (encoding + uploading) were selected according to the spatial and temporal resolutions of the content being encoded and transmitted. These target times are shown in Table 6-A for each resolution group. The selection was based on the range of total times (encoding plus uploading) obtained when encoding the selected sequences with a fixed QP in the same software platform used to run the performance experiments. Selecting target times associated with meaningful operation points is important since it allows testing the accuracy of the proposed method in terms of meeting the target time requirements. If the selected target times are too low, the proposed time control mechanism simply encodes the content using the highest possible QP (lowest quality, lowest bit rate) and is probably still unable to reach the target time. Conversely, very high target times do not challenge the proposed time control system since in this case, it simply selects the lowest QP allowed (highest quality, highest bit rate) throughout the whole sequence.

Regarding the structure of the encoded bit stream, the SOP structure used in the experiments reported in this section follows the RA configuration defined in the JCT

VC common test conditions [78] with a SOP size of 8 frames. The proposed scheme was designed mainly targeting HEVC video compression, even though there are no technical limitations that prevent it from being used in the context of a different hybrid video coding standard.

In terms of implementation, while the proposed scheme is applicable to any HEVC encoder implementation, due to the practical nature of the application, the implementation as well as experimental evaluation were all performed using a practical HEVC encoder. More research oriented implementations, such as the HEVC reference software [98], are generally not optimised for speed. The proposed method works by balancing encoding time as well as the uploading time and, as such, it is critical that a realistic encoding time is achieved by the chosen encoder. This is to avoid completely unbalancing the distribution of time, making the uploading time marginal. Therefore, the Turing codec [120] was selected as base for implementation, as this is an open source HEVC software encoder containing fast encoding presets and software optimisations that are essential in practical video compression applications [121]. The proposed method was implemented on top of the Turing codec (version 1.1) and all tests were run using the fast speed preset in single thread mode on Intel Xeon X3450 CPUs (2.67 GHz) with 8 GB of RAM.

6.4.2 Encoding times under fixed rate or constant quality conditions

As described in Section 6.2, state of the art rate control algorithms achieve the desired rate by continuously adapting the parameters that tune the rate-distortion decisions and the corresponding QP used in quantisation. Both these parameters can have a significant impact on the encoding time as well as the resulting quality of the sequence. Therefore, using rate control algorithms to fix the output bit rate and consequently limit the uploading time is not suitable for the scenarios targeted by the proposed method.

Some tests were performed to evaluate the impact of constant bit rate encoding on

quality as well as encoding time. Some sequences selected from the test set previously described in this section were encoded using the rate control algorithm adopted by the Turing codec [103] with the target bit rate indicated in Table 6-B. The results of these encodes can also be seen in Table 6-B. It can be observed that the encoding time is highly dependent on the target bit rate, where lower rates generally can be encoded faster than higher rates. In the case of the *FourPeople* sequence, encoding at 2 Mbps requires more than 60% additional time than encoding at 0.5 Mbps. Moreover, at a given bit rate (for the same resolution and frame rate) it can be observed that high variations can be expected in encoding time from sequence to sequence. As an example, it takes 516 seconds to encode the sequence *Kimono* at 2 Mbps, while only 383 seconds are needed to encode *ParkScene*. Finally, it can be observed that fixed bit rates produce different qualities across the tested sequences. An average PSNR of 40.13 dB was obtained for the sequence *Johnny*, against 37.13 dB for *FourPeople* when encoding at 0.5 Mbps.

Table 6-B: PSNR and encoding times for constant bit rate encoding.

Sequence	Target bit rate [kbps]	PSNR [dB]	Encoding time [s]
FourPeople	500	37.13	229.19
KristenAndSara	500	39.24	247.77
Johnny	500	40.13	222.60
FourPeople	2000	41.50	370.43
KristenAndSara	2000	42.54	392.18
Johnny	2000	42.59	377.56
Kimono	2000	38.37	516.25
ParkScene	2000	35.49	383.12
Kimono	4000	40.24	720.94
ParkScene	4000	37.37	513.99
Basketballdrive	4000	35.25	1170.19
Cactus	4000	35.39	977.61
Basketballdrive	8000	37.10	1458.55
Cactus	8000	36.79	1273.58

These figures show that using fixed rate conditions can produce significantly different outcomes in terms of the encoding time, as well as output quality. This highlights the fact that fixing the bit rate with a rate control method to control the uploading time has a significant impact on the encoding time, making it difficult to design a mechanism

to control the latter to meet the overall target time. The same set of sequences was also encoded under constant quality conditions, using fixed QP values ranging from 11 to 45. The results of these encodes are shown in Figure 6.7 in terms of average encoding time obtained for a sequence for each QP. The figure shows that the choice of different QPs lead to very different encoding times and that the encoding time is content dependent.

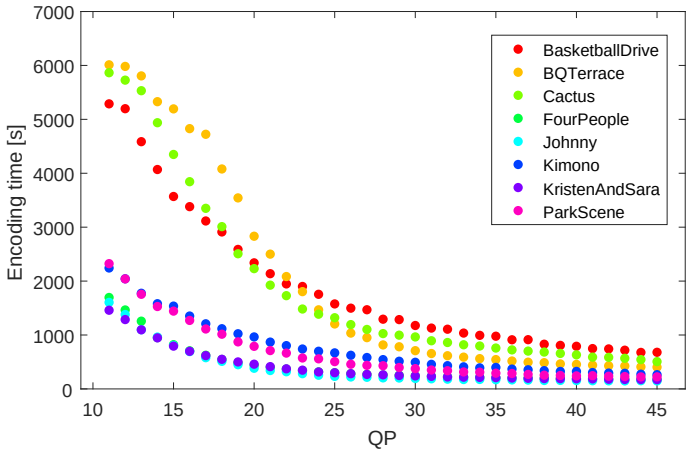
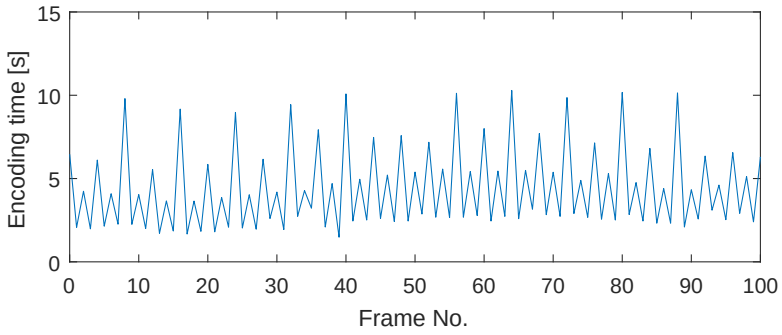
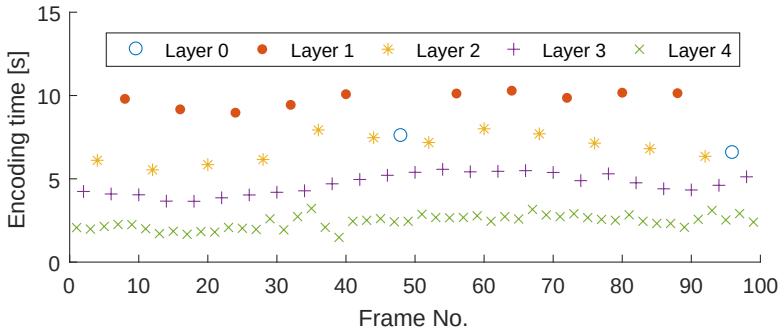


Figure 6.7: Average encoding times for different QP values.

The encoding time was also analysed in more detail on a frame by frame basis. The top plot in Figure 6.8 shows the encoding time per frame for the first 100 frames of the *BasketballDrive* sequence under constant QP conditions, for a QP value of 22. Significant differences can be observed in terms of encoding times from frame to frame. A breakdown of the encoding times for frames in each SOP layer is highlighted at the bottom of Figure 6.8. The difference in complexity is due to several factors, including the QP offset within the SOP and the different number of reference frames used in each SOP layer, which has an impact on the time necessary for inter predicting each block in the frame. Lower differences in encoding times can be observed for frames in the same SOP layer under constant QP conditions, as show in the bottom plot of Figure 6.8. This led to the design choice of performing the encoding time estimations independently for each SOP layer, as described in Subsection 6.3.2.



(a)



(b)

Figure 6.8: a) Encoding time variations for different frames in the *BasketballDrive* sequence under fixed QP conditions with the QP fixed to 22. b) Encoding times for frames in different SOP layers.

6.4.3 Accuracy of intermediate estimations

The tests performed to evaluate the performance of the proposed time control method consist of encoding all the selected test sequences with the respective target time constraints indicated in Table 6-A, for 3 different uploading bandwidths. The ultimate goal is to evaluate the accuracy of the proposed method in meeting the specified total target time, considering both encoding and uploading.

Tables 6-C and 6-D show the accuracy of the intermediate estimations performed by the proposed time control tool. In particular, the accuracy of the estimations of ρ, encoding time and total number of bits are shown separately for the different groups of selected test sequences. All accuracy values are presented in terms of the relative error

between the estimation and the real observed values in each case (ρ, encoding time or number of bits), computed as

$$err = \frac{|X - X'|}{X} \times 100 \tag{6.16}$$

where X and X' denote the real and the estimated values, respectively. All estimation errors reported in Tables 6-C and 6-D were computed for all SOPs in the encoded sequence apart from the first one, for which estimations are not available.

Table 6-C: Encoding time and total number of bits estimation errors.

Resolution	Overall bits estimation error	SOP-level bits estimation error	Overall time estimation error	SOP-level time estimation error
3840x2160@60fps	2.7%	7.3%	1.1%	9.0%
3840x2160@50fps	1.6%	8.4%	2.1%	9.3%
1920x1080@60fps	3.7%	11.7%	1.7%	10.6%
1920x1080@50fps	8.5%	13.1%	1.7%	13.3%
1920x1080@24fps	2.3%	7.3%	1.0%	9.8%
1280x720@60fps	4.4%	11.4%	3.4%	12.8%
1280x720@50fps	1.6%	7.5%	1.6%	10.0%
Overall	3.5%	9.5%	1.8%	10.7%

Table 6-D: ρ estimation errors.

Resolution	ρ_0	ρ_1	ρ_2	ρ_3	ρ_4
3840x2160@60fps	2.0%	3.9%	5.3%	5.5%	7.0%
3840x2160@50fps	3.6%	4.9%	6.3%	6.3%	10.1%
1920x1080@60fps	4.8%	6.7%	7.7%	7.4%	8.5%
1920x1080@50fps	5.9%	5.2%	8.1%	12.0%	14.8%
1920x1080@24fps	4.0%	3.6%	5.8%	8.6%	12.5%
1280x720@60fps	0.6%	3.2%	5.1%	6.7%	22.3%
1280x720@50fps	3.2%	4.5%	4.2%	3.9%	8.2%
Overall	3.4%	4.6%	6.1%	7.2%	11.9%

The overall bits estimation error column in Table 6-C shows the error of the number of bits estimated with respect to the number of bits used, computed after encoding the whole sequence. The SOP level bits estimation error column shows the average of the estimation errors computed SOP by SOP. Equivalent estimation errors are also

reported for the encoding time. It can be observed that the overall bits and encoding time estimations are very accurate, with an average error for all classes of 3.5% and 1.8%, respectively. The average SOP level error is slightly higher, around 10%, which is also an acceptable accuracy value. However, the higher SOP level error is not reflected in the overall error. This is because the proposed method performs on-the-fly decisions on the right QP to use after encoding each SOP and therefore is able to adapt to possible estimation errors made in the past. Also, it was observed that the SOP level error is in general higher for SOPs where a very low number of bits is used and small estimation errors result in high relative errors. These cases have a minor impact in the overall number of bits estimation since these SOPs contribute less to the overall number of bits spent. The same rationale applies to the encoding time prediction.

Finally, Table 6-D shows the estimation errors of the ρ values for different SOP layers. The five ρ estimation error columns in Table 6-D correspond to the average ρ estimation error computed according to Eq. (6.16) for all frames in the respective SOP layer. This error takes into account the ρ value that was estimated for a given frame in a given layer and the ρ value that was actually observed. The results in Table 6-D show that in the case of ρ estimations, the average estimation errors are, on average, below 12%. It can also be observed that the estimation errors are lower for frames in lower SOP layers, such as for layers 0 and 1 with erros of 3.4% and 4.6%. This can be explained by the fact that frames at higher SOP layers are encoded with higher QP offsets using reference frames that are temporally closer. This means that predictions are in general better and the amount of residual information is lower, leading to very low ratios of non zero levels and consequently degrading the reliability of the relative errors.

6.4.4 Overall performance

The overall performance of the proposed time control scheme is summarised in Table 6-E. In this table, the ability to accurately meet the total target time specified to the proposed method is measured using the relative error between the specified total target

time and the actual total time spent on encoding and uploading, as in Eq. (6.16). The proposed method is also compared with using a fixed QP to encode the whole length of each test sequence. For this purpose, all selected test sequences were encoded using all QPs ranging from 11 to 45 in a fixed QP configuration. For a given target time and uploading link bandwidth, the encoded bit stream that resulted in the best total encoding and uploading time with respect to each target time was then selected as the best fixed QP configuration for comparison. It is important to clarify that a fixed QP configuration means that the base QP selected for each SOP is the same throughout the sequence, although different QPs are used in frames belonging to different SOP layers. The QP offsets in different layers follow the typical RA configuration represented in Figure 6.2, which is also used in the proposed time control method. It is also important to note that this method cannot be used in practice since the ideal fixed QP value cannot be determined before the encoding process. This configuration can therefore be seen as the ideal fixed QP scenario, even though it is not applicable in the practical scenarios targeted by the proposed method.

The main aspect highlighted in the results reported in Table 6-E is the very low average accuracy error associated with the proposed solution. This supports the assumption that the average SOP level estimation errors reported in Table 6-C have a small impact in the overall accuracy of the proposed scheme. As previously explained, by operating on a SOP by SOP basis and readjusting the time left after encoding each SOP, the proposed time control algorithm is able to adjust to possible estimation errors that were made in previous SOPs. This allows selecting the right QPs for a specific encoding state that will adjust the encoding process in order to meet the specified target time.

Since the target of the proposed scheme is to encode a given sequence with the highest possible quality given certain bandwidth and time constraints, the quality difference in terms of PSNR between the ideal fixed QP configuration and the proposed method is also shown in Table 6-C. As expected, all values in the quality difference column in Table 6-C are positive, meaning that the quality of the proposed method is slightly

Table 6-E: Overall performance of the proposed time control scheme.

Bandwidth	Resolution	Proposed TC accuracy	Best fixed QP accuracy	Quality difference [dB]
128 kbps	3840x2160@60fps	0.2%	3.1%	0.47
	3840x2160@50fps	0.2%	4.1%	0.53
	1920x1080@60fps	0.4%	5.7%	0.25
	1920x1080@50fps	0.4%	3.1%	0.96
	1920x1080@24fps	0.8%	2.5%	0.80
	1280x720@60fps	0.3%	5.2%	0.33
	1280x720@50fps	0.5%	4.8%	0.39
256 kbps	3840x2160@60fps	0.3%	3.3%	0.56
	3840x2160@50fps	0.2%	3.1%	0.54
	1920x1080@60fps	0.3%	6.4%	0.39
	1920x1080@50fps	0.4%	3.2%	1.02
	1920x1080@24fps	0.7%	2.7%	0.81
	1280x720@60fps	0.4%	3.2%	0.38
	1280x720@50fps	0.2%	3.6%	0.63
512 kbps	3840x2160@60fps	0.2%	2.5%	0.61
	3840x2160@50fps	0.1%	3.9%	0.56
	1920x1080@60fps	0.3%	6.8%	0.29
	1920x1080@50fps	0.4%	2.9%	1.02
	1920x1080@24fps	0.5%	2.7%	0.83
	1280x720@60fps	0.4%	3.6%	0.41
	1280x720@50fps	0.2%	4.3%	0.77
Overall		**0.4%**	**3.6%**	**0.63**

lower than the one produced by the ideal fixed QP configuration. This is expected since the proposed method starts encoding a given sequence with a pre determined QP of 27 (approximately in the middle of the range of allowed QPs) and then needs to adapt to the specified time constraints during the encoding process. Nevertheless, the average quality difference for all sequences and bandwidths is only 0.63 dB, which highlights the ability of the proposed solution to produce comparable quality encoded bit streams with respect to an ideal case. This is also highlighted in Figure 6.9 where the output video quality measured with PSNR for the sequence *BQTerrace* is plotted as a function of the total time spent encoding and uploading with both methods. The output quality obtained with the proposed method increases as the specified total target time increases, as expected, and these qualities are similar to what is obtained with the ideal fixed QP

configuration. In some cases, the plot shows that for some target times, the video quality generated by the proposed method can even be slightly higher than the best fixed QP point. For example, for a target time of 6000 seconds and a bandwidth of 256 kbps, the best fixed QP point that produces a total time lower than the target time provides slightly lower quality than the proposed method. This is because using fixed QP does not provide any adaptation to the given target time. Overall, the plot reinforces the fact that the ability of adjusting to different constraints of the proposed method comes at the cost of a minor average PSNR drop, in general.

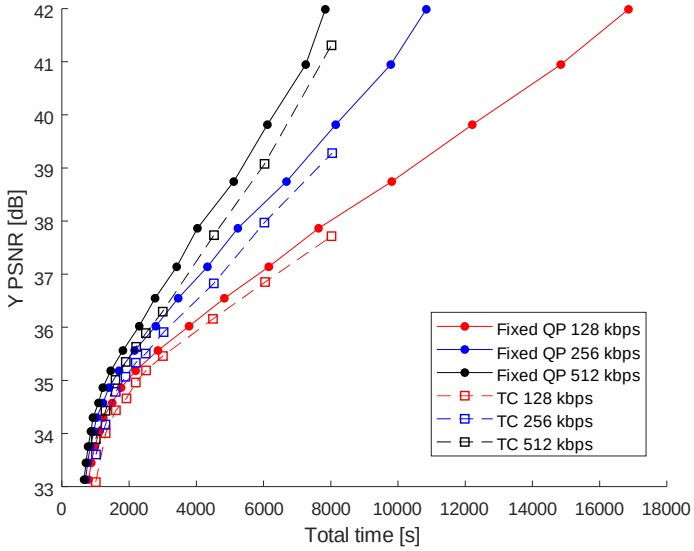


Figure 6.9: Output quality comparison between the proposed method and the ideal fixed QP approach for different target times and uploading bandwidths for the sequence BQTerrace.

In order to highlight the adaptation capabilities of the proposed tool, Figure 6.10 shows the QPs selected by the proposed time control scheme for each SOP. These results were obtained when encoding the sequence *RushHour* for different total target times considering an uplink bandwidth of 512 kbps. This figure gives a good insight of the ability of the proposed method to adapt to different time requirements. It shows that, after encoding the first SOP with the predefined QP of 27, different target times trigger the selection of different QPs. In each case, the encoding process eventually stabilises around a given QP value and performs on the fly adjustments if needed, according to

the ongoing encoding process.

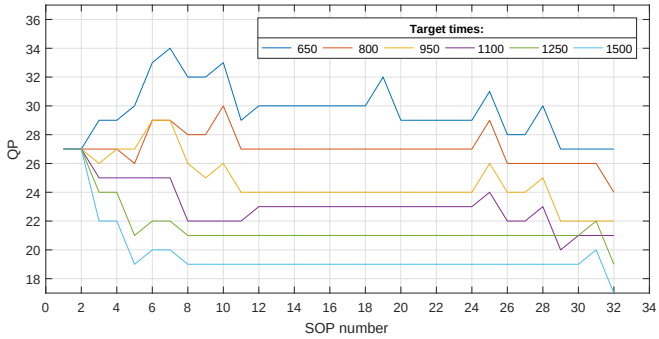


Figure 6.10: QPs selected when encoding the sequence *RushHour* considering a bandwidth of 512 kbps and 5 different total target times ranging from 500 seconds to 2000 seconds.

Finally, Figure 6.11 shows the encoding time and uploading time distribution observed when encoding 4 different 1920×1080 sequences considering 3 different transmission bandwidths. The bars represent the encoding and uploading times observed when encoding with a total target time of 1900 seconds. It can be seen that the proposed method is able to adapt to the specified uploading bandwidth, allocating more time to the encoding process for higher bandwidths. This is achieved by selecting lower QPs in these cases in order to maximise the quality of the output bit stream.

6.5 Conclusion

This chapter proposed a joint encoding and uploading time control scheme based on an adaptive QP selection algorithm that relies on accurate encoding time and bit rate estimation techniques performed during the encoding process. The overall scheme targets video compression applications where video content needs to be encoded and transmitted within given overall time constraints. The reported experimental results show that the proposed method is able to accurately meet the overall time constraints for different overall target times and different bandwidths considered for transmission. Future research work can be performed to refine the proposed technique, for example by defin-

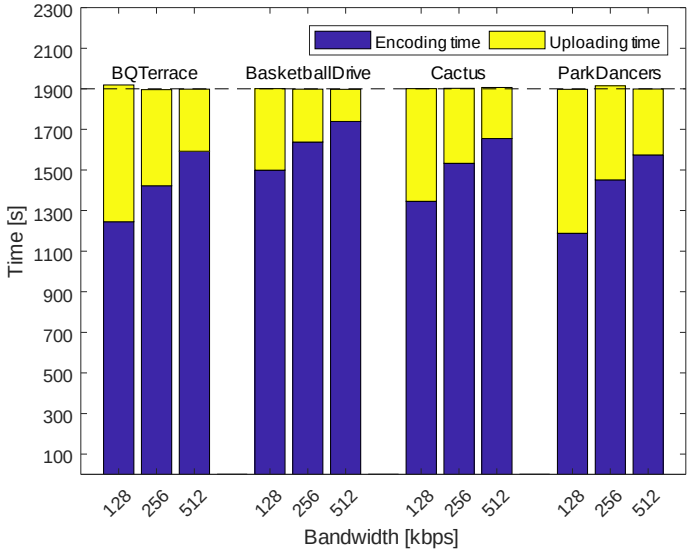


Figure 6.11: Encoding and uploading time distributions for different uploading bandwidths.

ing a method to select an appropriate initial base QP depending on the overall time constraints and characteristics of the video content.

Chapter 7

Conclusions and future work

The number of new services and devices that allow the creation, distribution and consumption of video content keeps increasing at an outstanding pace. This highly contributes to the continuous growth of the amount of multimedia information flowing in today's communication networks. This, together with the introduction of new and more immersive video formats, such as UHD resolutions or 360 degree video, increases the need for more efficient video compression techniques. Video applications and services becoming increasingly more popular include real-time streaming of high definition content over the internet, UHD broadcasting, screen mirroring among a variety of devices or high quality video recording and sharing. These are just a few examples of applications that are directly affected by the compression efficiency of the underlying video coding technology.

In this context, the aim of the research work presented in this thesis is to study, design and assess new video compression tools based on the state-of-the-art HEVC standard. In particular, the presented research work was divided into the following main areas of work:

1. The properties and limitations of the Human Visual System (HVS) were exploited to tune the performance of HEVC encoders towards a better subjective quality,

rather than optimising purely according to mathematical differences. This included the identification of areas of the video content where distortions are less noticeable by the HVS and therefore higher compression can be applied in these areas without noticeable perceptual quality degradations.

2. The prevention of contouring artefacts in compressed video was investigated in the context of HEVC video encoding. This type of artefacts can significantly degrade the perceptual quality of the decoded video sequences. The work in this area focused on providing a good insight on how these artefacts appear and how they can be prevented to minimise their impact in the particular case of UHD content.

3. New coding tools with compression capabilities beyond HEVC were investigated, in particular in the area of Intra coding. Different Intra prediction improvement techniques were studied and tools to improve HEVC's Intra coding performance were proposed.

4. The application of HEVC in practical video compression applications, where encoding time plays an important role, was studied in the context of joint encoding and uploading time control. A system comprising accurate encoding time and bit rate estimation techniques was designed in order to meet the overall processing time requirements of applications with overall encoding and uploading time restrictions.

According to the previously mentioned areas of research work, a brief summary of the major contributions presented in the thesis is reported in the following. These contributions are listed according to the chapter in which they were presented.

1. In Chapter 3, a novel technique for integrating a Just Noticeable Distortion (JND) model into an HEVC encoder was proposed, allowing a perceptually-oriented selection of the quantised levels by the Rate Distortion Optimised Quantisation process. The proposed technique relies on a JND model specifically adapted to work within a video encoder. The model is used to modify the decisions made by the RDOQ process at the encoder side, meaning that a fully compliant HEVC bit stream is

generated with the proposed solution. Given that the required extra complexity is very low, the proposed technique shows potential to be incorporated in practical HEVC video encoders in order to reduce the compressed bit rates without perceptual quality degradations. Particularly when targeting high video qualities, the tests performed show that this model can provide significant bit rate reductions without perceived quality degradations, which can possibly be a significant advantage for a practical video encoder with respect to its competitors.

2. In Chapter 4, two techniques based on the reduction of the Quantisation Parameter (QP) in areas prone to contouring artefacts were proposed to prevent these artefacts from appearing in UHD sequences after compression. The first technique consisted of increasing the video quality in these areas by performing quantisation with lower QPs in contouring-prone areas. The second technique consisted of an extension of the first method. This extension relies on modifying the RD costs associated with the merge mode candidates in Inter frames, aiming to further prevent the visibility of contouring artefacts in the decoded video sequences. Both solutions analysed are mainly targeted at higher bit rate scenarios, since for lower bit rates contouring artefacts may be considered less significant when compared to other visual degradations, as explained in Chapter 4.

3. In Chapter 5, two different approaches to improve the performance of Intra coding in HEVC were proposed. The first one consisted of applying predetermined spatial patterns, known both at the encoder and decoder, to compensate for the inaccuracies of Intra prediction blocks. Studying this approach and possible derivations mainly provided valuable knowledge on the characteristics of Intra coding residuals, as the obtained coding performance gains were marginal. The second approach was based on the Combined Intra Prediction (CIP) mechanism. An improved CIP technique was proposed to avoid mismatches in the CIP values computed at the encoder and decoder at the top and left borders of the prediction blocks. A combination of the proposed improved CIP technique with another Intra prediction

improvement technique, MPI, was also proposed. Both improved CIP variations show improved compression efficiency performance with respect to the reference HEVC software, as reported in Chapter 5.

4. In Chapter 6, a joint encoding and uploading time control system based on an adaptive QP selection algorithm was proposed. The first novelty introduced by this system is the proposed algorithm for joint encoding and uploading time control based on adaptive QP selection for groups of frames. To achieve an accurate performance of this algorithm, a novel technique to estimate encoding time variations with the QPs used for encoding a group of frames, based on the percentage of non zero quantised coefficients, ρ, was proposed. Similarly, a novel technique to reuse ρ estimations to accurately estimate bit rates for a group of frames, based on a piecewise interpolation/extrapolation method, was also proposed. The reported experimental results show that the proposed method is able to accurately meet the overall time constraints for different overall target times and different bandwidths considered for uploading.

All contributions presented in this thesis can be further refined and improved. In the context of Chapter 3, further bit rate reduction could be achieved by including an additional factor in the adopted JND model to account for possible temporal masking effects. This would lower the number of bits allocated to parts of the video where the perception of details is lower due to fast motion in the content.

In the context of Chapter 4, the proposed methods could possibly achieve better performance by refining the contouring areas detection mechanism, improving this way the accuracy of the contouring maps. Additionally, a method to automatically select the adequate contouring QPs according to the video content being encoded is also needed to integrate the proposed techniques into practical video encoding applications. Finally, since the technique was designed assuming fixed Intra period intervals, a possible extension to cope with different Intra periods would also strengthen the proposed technique in the context of practical applications.

In the context of Chapter 5, further studies on Intra prediction improvements using CIP may be of interest. This concept has not been adopted in any video compression standard so far but the performance results reported in this chapter for the proposed techniques show that video compression gains can be achieved with respect to HEVC. Understanding how most of these gains can be retained with lower complexity could make this technique more attractive for practical implementations, so that it could possibly be considered in future video coding standards.

Finally, in the context of Chapter 6, defining a method to select an appropriate initial base QP would strengthen the proposed joint time control algorithm, from a practical perspective. This selection could, for example, take into account a combination of the overall time constraints and the characteristics of the video content to encode.

List of publications

Journal papers

1. A. S. Dias, M. Naccari and M. Mrak, "Contouring artefacts prevention in compressed UHD video sequences: tools and analysis of their performance," *IEEE ComSoc MMTC Communications Frontiers*, vol. 11, no. 1, pp. 67-72, 2016.

2. A. S. Dias, S. Huang, S. G. Blasi, M. Mrak and E. Izquierdo, "Time-constrained video delivery using adaptive coding parameters," *IEEE Transactions on Circuits and Systems for Video Technology* (Accepted for publication).

Conference papers

1. A. S. Dias, S. Schwarz, M. Siekmann, S. Bosse, H. Schwarz, D. Marpe, J. Zubrzycki and M. Mrak, "Perceptually optimised video compression," *International Conference on Multimedia and Expo*, Turin, Italy, June 2015.

2. A. S. Dias, M. Siekmann, S. Bosse, H. Schwarz, D. Marpe and M. Mrak, "Rate-distortion optimised quantisation for HEVC using spatial just noticeable distortion," *European Signal Processing Conference*, Nice, France, August 2015.

3. A. S. Dias and M. Mrak, "Region-adaptive quantisation for prevention of contouring in coded video," *IEEE International Conference on Multimedia Signal Process-*

ing, Xiamen, China, October 2015.

4. A. S. Dias, S. G. Blasi, M. Mrak and E. Izquierdo "Improved combined Intra prediction for higher video compression efficiency," *Picture Coding Symposium*, Nuremberg, Germany, December 2016.

5. S. G. Blasi, A. S. Dias, M. Mrak, S. Huang and E. Izquierdo, "Complexity-constrained video encoding and delivery using configuration transfer matrix," *Picture Coding Symposium*, San Francisco, CA, USA, June 2018.

6. A. S. Dias, S. G. Blasi, M. Mrak, S. Huang and E. Izquierdo, "Adaptive video encoding for time-constrained compression and delivery," *European Signal Processing Conference*, Rome, Italy, September 2018 (Accepted).

7. A. S. Dias, S. G. Blasi, F. Rivera, E. Izquierdo and M. Mrak, "An overview of recent video coding developments in MPEG and AOMedia," *International Broadcasting Convention*, Amsterdam, Netherlands, September 2018 (Accepted).

Contributions to standardisation

1. A. S. Dias and M. Mrak, "Visualisation of BD-rate interpolation curves for MOS and PSNR results," JCT-VC, Warsaw, Poland, Tech. Rep. JCTVC-U0183, June 2015.

References

[1] G. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, 2012.

[2] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, 2003.

[3] D. Hubel, *Eye, brain and vision.* Scientific American Library, 1995.

[4] A. Roorda, "Human visual system - Image formation," in *The Encyclopedia of Imaging Science and Technology, Vol. 1*, J. Hornak, Ed. John Wiley & Sons, 2002, pp. 539–557.

[5] R. Hunt, *The reproduction of colour.* John Wiley & Sons, 2004.

[6] ITU-T, "ITU-T Recommendation BT.709 - Parameter values for the HDTV standards for production and international programme exchange," ITU-T, Tech. Rep., June 2015.

[7] ——, "ITU-T Recommendation BT.2020 - Parameter values for ultra-high definition television systems for production and international programme exchange," ITU-T, Tech. Rep., October 2015.

[8] ITU-R, "Image parameter values for high dynamic range television for use in production and international programme exchange," ITU-R, Tech. Rep., June 2017.

[9] C. Poynton, *Digital video and HD: algorithms and interfaces.* Morgan Kaufman Publishers, 2012.

[10] R. Salmon, M. Armstrong, and S. Jolly, "Higher frame rates for more immersive

video and television," BBC, Tech. Rep., September 2011.

[11] K. Noland, "High frame rate television: sampling theory, the human visual system, and why the Nyquist-Shannon theorem does not apply," *SMPTE Motion Imaging Journal*, vol. 125, no. 3, pp. 46–52, 2016.

[12] Z. Wang and A. Bovik, "Mean squared error: love it or leave it? A new look at signal fidelity measures," *IEEE Signal Processing Magazine*, vol. 26, no. 1, pp. 98–117, 2009.

[13] C. Lambrecht, *Vision models and applications to image and video processing.* Springer, 2001.

[14] H. Wu and K. Rao, *Digital video image quality and perceptual coding.* CRC Press, 2006.

[15] Z. Wang, E. Simoncelli, A. Bovik, and S. Hamid, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[16] M. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Transactions on Broadcasting*, vol. 50, no. 3, pp. 312–322, 2004.

[17] ITU-T, "Objective perceptual video quality measurement techniques for digital cable television in the presence of a full reference," ITU-T, Tech. Rep. ITU-T Recommendation J.144, 2004.

[18] ITU-R, "Objective perceptual video quality measurement techniques for standard definition digital broadcast television in the presence of a full reference," ITU-R, Tech. Rep. ITU-R Recommendation BT.1683, 2004.

[19] ANSI, "Digital transport of one-way video signals - Parameters for objective performance assessment," ANSI, Tech. Rep. ANSI T1.801.03, 2003.

[20] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electronics Letters*, vol. 44, no. 13, pp. 800–801, 2008.

[21] G. Sullivan and T. Wiegand, "Rate-distortion optimization for video compression," *IEEE Signal Processing Magazine*, vol. 15, no. 6, pp. 74–90, 1998.

[22] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves,"

VCEG, Austin, TX, USA, Tech. Rep. VCEG-M33, April 2001.

[23] ITU-T, "ITU-T Recommendation H.261: Video codec for audiovisual services at p×64 kbit/s," ITU-T, Tech. Rep. ITU-T Recommendation H.261, 1988.

[24] J. Ohm, *Multimedia communication technology*. Springer, 2004.

[25] D. Flynn, D. Marpe, M. Naccari, T. Nguyen, C. Rosewarne, K. Sharman, J. Sole, and J. Xu, "Overview of the range extensions for the HEVC standard: tools, profiles, and performance," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 4–19, 2016.

[26] G. Tech, Y. Chen, K. Müller, J. Ohm, A. Vetro, and Y.-K. Wang, "Overview of the Multiview and 3D Extensions of High Efficiency Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 35–49, 2016.

[27] J. Xu, R. Joshi, and R. Cohen, "Overview of the emerging HEVC Screen Content Coding extension," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 50–62, 2016.

[28] VCEG, "Draft requirements for next generation video coding project," VCEG, London, UK, Tech. Rep. VCEG-AL96, July 2009.

[29] T. Tan, R. Weerakkody, M. Mrak, N. Ramzan, V. Baroncini, J. Ohm, and G. Sullivan, "Video quality evaluation methodology and verification testing of HEVC compression performance," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 1, pp. 76–90, 2016.

[30] D. Taubman and M. Marcellin, *JPEG 2000: Image compression fundamentals standards and practice*. Springer, 2002.

[31] K. Ugur and J. Lainema, "Updated results on HEVC still picture coding performance," JCT-VC, Incheon, South Korea, Tech. Rep. JCTVC-M0041, April 2012.

[32] R. Weerakkody and M. Mrak, "High efficiency video coding for Ultra High Definition Television," in *NEM Summit*, Nantes, France, October 2013.

[33] J. Lainema, F. Bossen, W. Han, J. Min, and K. Ugur, "Intra coding of the HEVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 17 926–1801, 2012.

[34] Y. Zheng, M. Coban, and M. Karczewicz, "Simplified Intra smoothing," JCT-VC, Guangzhou, China, Tech. Rep. JCTVC-C234, October 2010.

[35] B. Girod, "Motion-compensating prediction with fractional-pel accuracy," *IEEE Transactions on Communications*, vol. 41, no. 4, pp. 604–612, 1993.

[36] K. Ugur, A. Alshin, E. Alshina, F. Bossen, W.-J. Han, J.-H. Park, and J. Lainema, "Interpolation filter design in HEVC and its coding efficiency - complexity analysis," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Vancouver, BC, Canada, May 2013.

[37] K. Ugur, A. Alshin, E. Alshina, F. Bossen, W. Han, J. Park, and J. Lainema, "Motion compensated prediction and interpolation filter design in H.265/HEVC," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 6, pp. 946–956, 2013.

[38] P. Helle, S. Oudin, B. Bross, D. Marpe, M. Bici, K. Ugur, J. Jung, G. Clare, and T. Wiegand, "Block merging for qadtree-based partitioning in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1720–1731, 2012.

[39] J. Lin, Y. Chen, Y. Tsai, Y. Huang, and S. Lei, "Motion vector coding techniques for HEVC," in *IEEE International Workshop on Multimedia Signal Processing*, Hangzhou, China, October 2011.

[40] W.-J. Han, J. Min, I. Kim, E. Alshina, A. Alshin, T. Lee, J. Chen, V. Seregin, S. Lee, Y. Hong, M. Cheon, N. Shlyakhov, K. McCann, T. Davies, and J. Park, "Improved video compression efficiency through flexible unit representation and corresponding extension of coding tools," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 12, pp. 1709–1720, 2010.

[41] T. Nguyen, P. Helle, M. Winken, B. Bross, D. Marpe, H. Schwarz, and T. Wiegand, "Transform coding techniques in HEVC," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 6, pp. 978–989, 2013.

[42] V. Britanak, P. Yip, and K. Rao, *Discrete cosine and sine transforms: general properties, fast algorithms and integer approximations.* Academic Press, 2006.

[43] A. Saxena and F. Fernandes, "DCT/DST-based transform coding for Intra pre-

diction in image/video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 10, pp. 3974–3981, 2013.

[44] K. Ugur and A. Saxena, "CE1: Summary report on Core Experiment on Intra transform mode dependency simplifications," JCT-VC, Stockholm, Sweden, Tech. Rep. JCTVC-J0021, July 2012.

[45] K. Ugur and O. Bici, "Performance evaluation of DST in intra prediction," JCT-VC, Geneva, Switzerland, Tech. Rep. JCTVC-I0582, May 2012.

[46] A. Gabriellini, M. Naccari, M. Mrak, and D. Flynn, "Spatial transform skip in the emerging High Efficiency Video Coding standard," in *IEEE International Conference on Image Processing*, Orlando, FL, USA, October 2012.

[47] M. Budagavi, A. Fuldseth, G. Bjøntegaard, V. Sze, and M. Sadafale, "Core transform design in the High Efficiency Video Coding (HEVC) standard," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 6, pp. 1029–1041, 2013.

[48] J. Sole, R. Joshi, N. Nguyen, T. Ji, M. Karczewicz, G. Clare, F. Henry, and A. Duenas, "Transform coefficient coding in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1765–1777, 2012.

[49] D. Marpe, H. Schwarz, and T. Wiegand, "Context based adaptive binary arithmetic coding in the H.264/AVC video compression standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 620–636, 2003.

[50] T. Chen, Y. Huang, C. Tsai, B. Hsieh, and L. Chen, "Architecture design of context-based adaptive variable-length coding for H.264/AVC," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 9, pp. 832–836, 2006.

[51] V. Sze and M. Budagavi, "High throughput CABAC entropy encoding in HEVC," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1778–1791, 2012.

[52] C. Fu, E. Alshina, A. Alshin, Y. Huang, C. Chen, C. Tsai, C. Hsu, S. Lei, J. Park, and W. Han, "Sample adaptive offset in the HEVC standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1755–1764, 2012.

[53] A. Norkin, G. Bjontegaard, A. Fuldseth, M. Narroschke, M. Ikeda, K. Andersson, M. Zhou, and G. Auwera, "HEVC deblocking filter," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1746–1754, 2012.

[54] K. Misra, A. Segall, M. Horowitz, S. Xu, A. Fuldseth, and M. Zhou, "An overview of tiles in HEVC," *IEEE Journal of Selected Topics in Signal Processing*, vol. 7, no. 6, pp. 969–977, 2013.

[55] C. Chi, M. Alvarez-Mesa, B. Juurlink, G. Clare, F. Henry, S. Pateux, and T. Schierl, "Parallel scalability and efficiency of HEVC parallelization approaches," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1827–1838, 2012.

[56] R. Sjoberg, Y. Chen, A. Fujibayashi, M. Hannuksela, J. Samuelsson, T. Tan, Y. Wang, and S. Wenger, "Overview of HEVC high-level syntax and reference picture management," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1858–1870, 2012.

[57] M. Wien, *High Efficiency Video Coding (HEVC) - Algorithms and architectures.* Springer, 2014.

[58] V. Sze, M. Budagavi, and G. Sullivan, *High Efficiency Video Coding - Coding tools and specification.* Springer, 2014.

[59] F. Nes and M. Bouman, "Spatial modulation transfer in the human eye," *Journal of the Optical Society of America*, vol. 57, no. 3, pp. 401–406, 1967.

[60] A. Ahumada and H. Peterson, "Luminance-model-based DCT quantization for color image compression," in *Human Vision, Visual Processing, and Digital Display III*, San Jose, CA, USA, August 1992.

[61] A. Watson, "DCTune: A technique for visual optimization of DCT quantization matrices for individual images," *Society for Information Display Digest of Technical Papers XXIV*, vol. 24, pp. 946–949, 1993.

[62] X. Zhang, W. Lin, and P. Xue, "Improved estimation for just-noticeable visual distortion," *Signal Processing*, vol. 85, no. 4, pp. 795–808, 2005.

[63] Y. Jia, W. Lin, and A. Kassim, "Estimating just-noticeable distortion for video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 7,

pp. 820–829, 2006.

[64] X. Yang, W. Lin, Z. Lu, E. Ong, and S. Yao, "Motion-compensated residue preprocessing in video coding based on just-noticeable-distortion profile," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 6, pp. 742–752, 2005.

[65] C. Chou and Y. Li, "A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, no. 6, pp. 467–476, 1995.

[66] C.-M. Mak and K. Ngan, "Enhancing compression rate by just-noticeable distortion model for H.264/AVC," in *IEEE International Symposium on Circuits and Systems*, Taipei, Taiwan, May 2009.

[67] Z. Chen and C. Guillemot, "Perceptually-friendly H.264/AVC video coding based on foveated just-noticeable-distortion model," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 6, pp. 806–819, 2010.

[68] M. Naccari and F. Pereira, "Advanced H.264/AVC-based perceptual video coding: architecture, tools, and assessment," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 6, pp. 766–782, 2011.

[69] Z. Wei and K. Ngan, "Spatio-temporal just noticeable distortion profile for grey scale image/video in DCT domain," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 3, pp. 337–346, 2009.

[70] M. Naccari and F. Pereira, "Integrating a spatial just noticeable distortion model in the under development HEVC codec," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Prague, Czech Republic, May 2011.

[71] M. Naccari and M. Mrak, "Intensity dependent spatial quantization with application in HEVC," in *IEEE International Conference on Multimedia and Expo*, San Jose, CA, USA, July 2013.

[72] J. Kim, S. Bae, and M. Kim, "An HEVC-compliant perceptual video coding scheme based on JND models for variable block-sized transform kernels," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 11, pp. 1786–1800, 2015.

[73] S. Bae and M. Kim, "A novel generalized DCT-based JND profile based on an elaborate CM-JND model for variable block-sized transforms in monochrome images," *IEEE Transactions on Image Processing*, vol. 23, no. 8, pp. 3227–3240, 2014.

[74] A. Watson and A. Ahumada, "A standard model for foveal detection of spatial contrast," *Journal of Vision*, vol. 5, no. 9, pp. 717–740, 2005.

[75] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[76] J. Foley and G. Boynton, "New model of human luminance pattern vision mechanisms: analysis of the effects of pattern orientation, spatial phase, and temporal frequency," in *Proceedings SPIE 2054, Computer Vision Based on Neurobiology*, Park Grove, CA, USA, March 1994.

[77] M. Karczewicz, Y. Ye, and I. Chong, "Rate distortion optimized quantization," VCEG, Antalya, Turkey, Tech. Rep. VCEG-AH21, January 2008.

[78] F. Bossen, "Common HM test conditions and software reference configurations," JCT-VC, Geneva, Switzerland, Tech. Rep. JCTVC-L1100, January 2013.

[79] W. Ahn and J.-S. Kim, "Flat-region detection and false contour removal in the digital TV display," in *IEEE International Conference on Multimedia and Expo*, Amsterdam, The Netherlands, July 2005.

[80] S. Bhagavathy, J. Llach, and J. Zhai, "Multiscale probabilistic dithering for suppressing contour artifacts in digital images," *IEEE Transactions on Image Processing*, vol. 18, no. 9, pp. 1936–1945, 2009.

[81] Y. Wang, C. Abhayaratne, R. Weerakkody, and M. Mrak, "Multi-scale dithering for contouring artefacts removal in compressed UHD video sequences," in *IEEE Global Conference on Signal and Information Processing*, Atlanta, GA, USA, December 2014.

[82] J. Lee, B. Lim, R. Park, J.-S. Kim, and W. Ahn, "Two-stage false contour detection algorithm using re-quantization and directional contrast features and its application to adaptive false contour reduction," in *International Conference on Consumer Electronics*, Las Vegas, NV, USA, January 2006.

[83] K. Yoo, H. Song, and K. Sohn, "In-loop selective processing for contour artefact

reduction in video coding," *Electronics Letters*, vol. 45, no. 20, pp. 1020–1022, 2009.

[84] T. Tan and Y. Suzuki, "Contouring artefact and solution," JCT-VC, Shanghai, China, Tech. Rep. JCTVC-K0139, October 2012.

[85] N. Casali, M. Naccari, M. Mrak, and R. Leonardi, "Adaptive quantisation in HEVC for contouring artefacts removal in UHD content," in *IEEE International Conference on Image Processing*, Quebec, Canada, September 2015.

[86] A. Gabriellini, D. Flynn, M. Mrak, and T. Davies, "Combined Intra-prediction for High-Efficiency Video Coding," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 7, pp. 1282–1289, 2011.

[87] Y. Ye and M. Karczewicz, "Improved H.264 Intra coding based on bi-directional Intra prediction, directional transform, and adaptive coefficient scanning," in *IEEE International Conference on Image Processing*, San Diego, CA, USA, October 2008.

[88] C. Yeh, T. Tseng, C. Lee, and C. Lin, "Predictive texture synthesis-based Intra coding scheme for Advanced Video Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 9, pp. 1508–1514, 2015.

[89] T. Tan, C. S. Boon, and Y. Suzuki, "Intra prediction by template matching," in *IEEE International Conference on Image Processing*, Atlanta, GA, USA, March 2006.

[90] Y. Guo, Y. K. Wang, and H. Li, "Priority-based template matching Intra prediction," in *IEEE International Conference on Multimedia and Expo*, Hannover, Germany, March 2008.

[91] S. Cherigui, C. Guillemot, D. Thoreau, P. Guillotel, and P. Perez, "Correspondence map-aided neighbor embedding for image Intra prediction," *IEEE Transactions on Image Processing*, vol. 22, no. 3, pp. 1161–1174, 2013.

[92] D. Doshkov, P. Ndjiki-Nya, H. Lakshman, M. Koeppel, and T. Wiegand, "Towards efficient Intra prediction based on image inpainting methods," in *Picture Coding Symposium*, Nagoya, Japan, December 2010.

[93] M. Budagavi and D.-K. Kwon, "AHG8: Video coding using Intra motion compensation," JCT-VC, Incheon, South Korea, Tech. Rep. JCTVC-M0350, April

2013.

[94] H. Chen, Y. Chen, M. Sun, A. Saxena, and M. Budagavi, "Improvements on Intra block copy in natural content video coding," in *IEEE International Symposium on Circuits and Systems*, Lisbon, Portugal, May 2015.

[95] S. Blasi, M. Mrak, and E. Izquierdo, "Frequency-domain Intra prediction analysis and processing for high-quality video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 5, pp. 798–811, 2015.

[96] J. Han, V. Melkote, and K. Rose, "Transform-domain temporal prediction in video coding: Exploiting correlation variation across coefficients," in *IEEE International Conference on Image Processing*, Hong Kong, China, September 2010.

[97] E. Alshina, A. Alshin, J.-H. Min, K. Choi, A. Saxena, and M. Budagavi, "Known tools performance investigation for next generation video coding," VCEG, Warsaw, Poland, Tech. Rep. VCEG-AZ05, June 2015.

[98] HM reference software. [Online]. Available: https://hevc.hhi.fraunhofer.de/HMdoc/

[99] K. Lim, G. Sullivan, and T. Wiegand, "Text description of joint model reference encoding methods and decoding concealment methods," JVT, Hong Kong, China, Tech. Rep. JVT-N046, January 2005.

[100] T. Chiang and Y. Zhang, "A new rate control scheme using quadratic rate distortion model," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 246–250, 1997.

[101] H. Choi, J. Nam, J. Yoo, D. Sim, and I. V. Bajic, "Rate control based on unified RQ model for HEVC," JCT-VC, San Jose, CA, USA, Tech. Rep. JCTVC-H0213, February 2012.

[102] X. Li, N. Oertel, A. Hutter, and A. Kaup, "Laplace distribution based Lagrangian rate distortion optimization for hybrid video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 2, pp. 193–205, 2009.

[103] B. Li, H. Li, L. Li, and J. Zhang, "Lambda domain rate control algorithm for high efficiency video coding," *IEEE Transactions on Image Processing*, vol. 23, no. 9, pp. 3841–3854, 2014.

[104] Z. He, Y. Kim, and S. Mitra, "Low delay rate control for DCT video coding via rho domain source modeling," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 8, pp. 928–940, 2001.

[105] ITU-T, "ITU-T recommendation H.263: Video coding for low bit rate communication," ITU-T, Tech. Rep. ITU-T Recommendation H.263, 1996.

[106] M. Liu, Y. Guo, H. Li, and C. Chen, "Low complexity rate control based on ρ domain for SVC," in *IEEE International Conference on Image Processing*, Hong Kong, China, September 2010.

[107] F. Bossen, B. Bross, K. Sühring, and D. Flynn, "HEVC complexity and implementation analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1685–1696, 2012.

[108] S. Cho and M. Kim, "Fast CU splitting and pruning for suboptimal CU partitioning in HEVC Intra coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 9, pp. 1555–1564, 2013.

[109] S. Ahn, B. Lee, and M. Kim, "A novel fast CU encoding scheme based on spatiotemporal encoding parameters for HEVC inter coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 3, pp. 422–435, 2015.

[110] G. Correa, P. Assuncao, L. Agostini, and L. Cruz, "Fast HEVC encoding decisions using data mining," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 4, pp. 660–673, 2015.

[111] A. Heindel, T. Haubner, and A. Kaup, "Fast CU split decisions for HEVC inter coding using support vector machines," in *Picture Coding Symposium*, Nuremberg, Germany, December 2016.

[112] J. Vanne, M. Viitanen, and T. D. Hamalainen, "Efficient mode decision schemes for HEVC inter prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 9, pp. 1579–1593, 2014.

[113] S. Blasi, I. Zupancic, E. Izquierdo, and E. Peixoto, "Adaptive precision motion estimation for HEVC coding," in *Picture Coding Symposium*, Cairns, Australia, May 2015.

[114] I. Zupancic and E. Izquierdo, "Fast motion estimation based on neighbouring cost

similarity," in *Picture Coding Symposium*, Nuremberg, Germany, December 2016.

[115] X. Li, M. Wien, and J. Ohm, "Rate-complexity-distortion optimization for hybrid video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 7, pp. 957–970, 2011.

[116] L. Merrit. x264: A high performance H.264/AVC encoder. [Online]. Available: http://akuvian.org/src/x264/overview_x264_v8_5.pdf

[117] G. Correa, P. Assuncao, L. Agostini, and L. Cruz, "Pareto-based method for high efficiency video coding with limited encoding time," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 9, pp. 1734–1745, 2016.

[118] ——, "Performance and computational complexity assessment of high efficiency video encoders," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1899–1909, 2012.

[119] J. Vanne, M. Viitanen, T. Hamalainen, and A. Hallapuro, "Comparative rate distortion complexity analysis of HEVC and AVC video codecs," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1885–1898, 2012.

[120] Turing codec software repository. [Online]. Available: https://github.com/bbc/turingcodec

[121] S. Blasi, M. Naccari, R. Weerakkody, J. Funnell, and M. Mrak, "The open-source turing codec: towards fast, flexible and parallel HEVC encoding," in *International Broadcasting Convention*, Amsterdam, Netherlands, September 2016.